2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Topology Matching of Branched Deformable Linear Objects

Manuel Zürn*, Markus Wnuk, Armin Lechler, Alexander Verl
Institute for Control Engineering of Machine Tools and Manufacturing Units,
University of Stuttgart, Seidenstr. 36, 70174 Stuttgart, Germany
(*e-mail: manuel.zuern@isw.uni-stuttgart.de)

***Abstract*—This paper presents a new method for correspondence estimation between a previously known topology of a branched deformable linear object and an image representation from a 3D stereo camera. Although frequently encountered in production, robotic deformable linear object manipulation still lacks reliable sensor feedback. Especially for branched deformable linear objects, such as wire harnesses, correspondence estimation is very challenging. Due to their flexible nature, they have an infinite-dimensional configuration space, such that visual appearances of the same object can vary strongly. Knowing the correspondence is vital for various applications, e.g., estimating valid grasping positions for robotic wire routing or augmented reality support for workers.**

Therefore, this paper presents a method for matching the topology of a branched deformable linear object to camera sensor data. Asymmetries in the wire harness design reduce the solution space by comparing the known topology of a model to the topology extracted from sensor data. The problem of finding the most likely solution to the matching problem requires features extracted from camera images. These features are used to construct a graph-based topology representation, which can then be matched to a graph-based topology representation of the known branched deformable linear object. The presented method is evaluated using multiple different non-overlapping configurations of a wire harness, showing the effectiveness of a graph-based segment matching approach.

***Keywords*—branched deformable linear object, wire harness localization, graph-based topology matching**

I. Introduction

Branched deformable linear objects (BDLOs), such as wire harnesses in Fig. 1, are frequently encountered in the industry to flexibly connect multiple components. Their flexibility makes them resilient against deviations between the components, but also challenging for manual or automatic manipulation, as they change their shape continuously during manipulation.

A particular example is an automotive wire harness, which has increased in size over the last decades due to various additional electrical functionality. This makes them a more costly component, not only in wire harness assembly but also in final installation in the car body.

The research leading to this publication has received funding from the German Research Foundation (DFG) as part of the International Research Training Group "Soft Tissue Robotics" (GRK 2198/1). Funded by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany´s Excellence Strategy – EXC 2075 – 390740016.
The authors would like to thank the Ministry of Science, Research and Arts of the Federal State of Baden-Württemberg for the financial support of the projects within the InnovationsCampus Future Mobility (ICM).

Automation in the BDLO field still lacks reliant sensor feedback solutions, as correspondence estimation between the infinite-dimensional configuration space with sensor data is still an open research question. Correspondence estimation can be used to estimate the next branch of the BDLO in a manipulation sequence. Therefore, it can not only be used for robotic manipulation but also assist workers in an augmented reality setup.

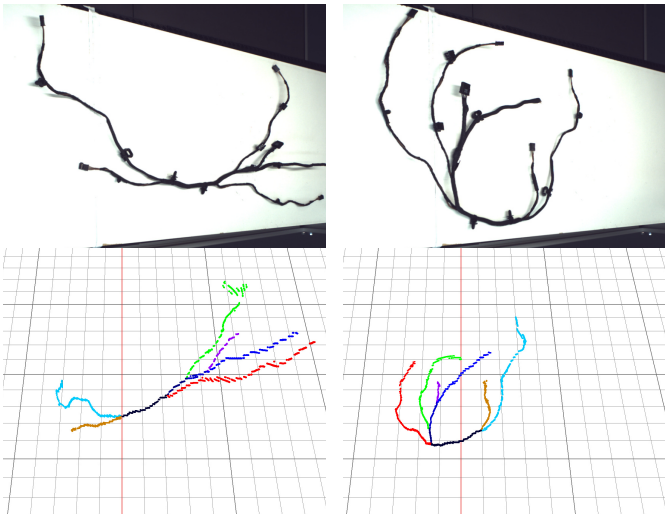


Figure 1. Topology matching for a wire harness in visual different configurations with a color-wise point cloud encoding.

Although sensor-based robotic manipulation is very common in the industry, deformable manipulation still lacks reliant sensor feedback solutions [1]. Extensive research has been done on deformable linear object (DLO) perception, modeling, and manipulation in recent years. However, current approaches for BDLO perception do not offer resilient solutions. One particular challenge towards this goal is to find a mapping between camera data and a BDLO model. This problem is referred to as correspondence estimation. For BDLOs, this leads to the problem of identifying a unique 1-1 correspondence between an image pixel and the corresponding part on the BDLO. This problem is referred to as point set registration, in the case of deformable objects as non-rigid point set registration. An overview of methods estimating non-rigid point-to-point correspondences can be found in [2]. This paper proposes a graph-based point-to-branch correspondence method that finds the correspondence by matching 3D camera

sensor data to a BDLO topology model and its geometry, see Fig. 1.

A. Related works

Few studies directly deal with BDLO localization using cameras. Therefore, this section is structured in a way to categorize the presented contribution in the field of robotic DLO and BDLO manipulation.

Wire harnesses are created by solving the wire harness routing problem [3], [4], where the topology of the wire harness is determined by solving an optimization problem to connect the different connectors. As a result of solving the routing problem, one obtains a topology of the wire harness, which usually shows the wire harness in a stretched configuration showing branch nodes, leaf nodes and the corresponding segment lengths [3]–[5]. This stretched representation can be used e.g., for testing the correct functionality of the wire harness [5].

Sensor-based feedback is needed for robotic manipulation of the wire harness. A popular approach in robotic DLO manipulation is to use 2D or 3D camera sensor feedback [6]–[9]. BDLO open further problems in perception, as they consist of multiple DLOs also called branches or segments. The first approach uses markers for branch identification of a BDLO and topology [10]. This approach has several disadvantages. It actively changes the BDLO in attaching markers fixed to the segments and is therefore more expensive in production. Furthermore, it is very sensitive to the orientation of the marker.

Markerless non-rigid point set registration is also used for correspondence estimation between sensor data and a discretized representation of a BDLO [8]. Point set registration requires segmented sensor data of the wire harness, as well as information of a previous configuration, and can therefore be used with high-frequency sensor feedback for real-time robotic control. DLO image segmentation has been dealt with extensively in recent publications [11], [12]. Instead of using the segmented image directly for point set registration, the idea of this paper is to do an additional preprocessing step of the input data, which gives information about the data point-to-branch connection. Similar approaches have already been successfully used to detect overlapping DLOs in [13]. This approach extends the idea of using branch and leaf nodes for BDLO and makes use of the topology from the cable harness routing problem, which is then used for the correspondence estimation between topological segment and segmented camera data.

The results of the topological matching can then be either used for direct component position estimation, as e.g., investigated in [14], or as additional information for non-rigid point set registration methods and robotic BDLO control.

B. Contribution

This paper introduces a way to use the topological representation of a wire harness for segment correspondence estimation. To do this, we create a directed graph out of the topology of the BDLO and match it with an undirected graph from nodes, which we extracted from a camera image of the real wire harness. After that, we introduce the matching problem, showing why asymmetry is useful and how to determine the topological uniqueness of a wire harness for different topologies. In an experiment, we evaluate the performance of the correspondence estimation and discuss afterwards further research questions, how the presented topological matching can be extended and how it can be used for various scenarios.

II. Topology of a branched deformable linear object

Let $T = (\mathcal{N}^{k+1}, \mathcal{E}^k, \mathcal{A}^k)$ represent the topology of a wire harness, where $\mathcal{N}$ represents a set of nodes with parents and children, $\mathcal{E}$ a set of segments, k being the number of segments and $\mathcal{A}$ a set of features. In industrial use cases, T can be extracted from the CAD modeling of the BDLO. Each segment $s \in \mathcal{E}$ is described by its parent and child node, as well as its corresponding feature set. Let $G = (\hat{\mathcal{N}}^{j+1}, \hat{\mathcal{E}}^j, \hat{\mathcal{A}}^j)$ represent an estimated undirected graph extracted from an BDLO, where $\hat{\mathcal{N}}$ represents estimated branch or leaf nodes, $\hat{\mathcal{E}}$ represents estimated segments and $\hat{\mathcal{A}}$ represents estimated features. $\hat{(\cdot)}$ denotes if the value has to be estimated from sensor data. Possible topological correspondences between the two graphs are $\mathcal{X}$, which match the segments of the directed graph to the segments of the undirected graph $\mathcal{X} \subset \mathcal{E}^k \times \hat{\mathcal{E}}^j$. $\mathcal{X}$ is represented as a binary correspondence matrix $\mathbf{X} \in \{0,1\}^{k \times j}$. In general, $j \leq k$ means that the number of nodes (and segments) in the undirected graph G can be less than the number of nodes (and segments) in the topology T. This results in finding a partial correspondence of G in T.

The sparse correspondence matrices $\mathbf{X}_i$, $\forall i = 1..m$ list all m possible solutions to the topological matching problem between the directed graph and the undirected graph. Let $f : \mathcal{X} \rightarrow \mathbb{R}$ be a function which minimizes the difference in features $\mathcal{A}$ between T and G as follows:

$$\arg\min_{\mathbf{X}} f(\mathbf{X}) = \|\mathcal{A} - \mathbf{X}\hat{\mathcal{A}}\|_{\mathbf{Q}}, \tag{1}$$

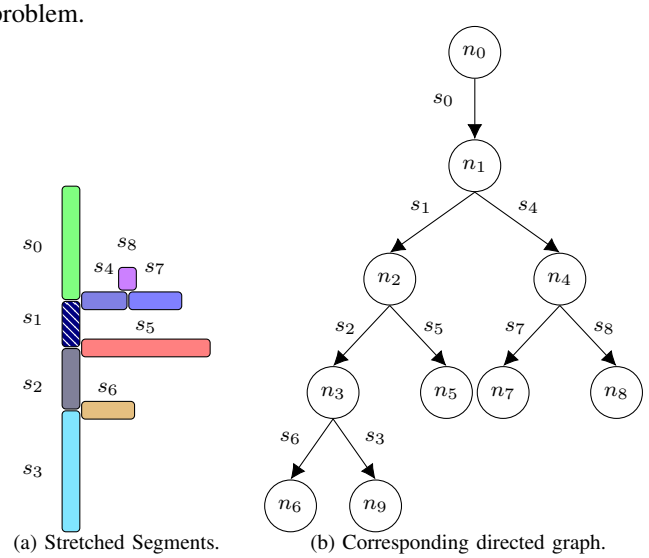
where $\mathbf{Q} \in \mathbb{R}^{h \times h}$ represents a feature weighting matrix for h different features in $\mathcal{A}$. Then the optimal mapping from G to T can be found by determining the minimum of $f(\mathbf{X}_i)$.

A. Directed graph

Solving the wire harness routing problem offers the topology of the BDLO. The topology T consists of k segments $\mathcal{E}$ as depicted in Fig. 2. The first node $n_0 \in \mathcal{N}$ has no parents, whereas all following nodes $\forall n \in \mathcal{N} \backslash n_0$ have exactly one parent node.

The segments have individual features, which can be the stretched length l_k, the radius $r_k(s)$, the color $c_k(s)$, plugs, wire channels, fixations, or sheats (features inspired by [15]), when $s \in [0..l_k]$ determines the local frame coordinate of the segment starting at the parent node and ending at the child node. Note that the list of features is not complete, it just illustrates some possibilities which can be used for the

feature matching problem. Either parent or child in Fig. 2 node shares the same index as the corresponding segment, depending on the directed graph. The directed graph can be either shown with nodes or with segments, where the segment representation improves readability for comparing it with the stretched segment representation in Fig. 2a, whereas the node representation improves the understanding of the matching problem.



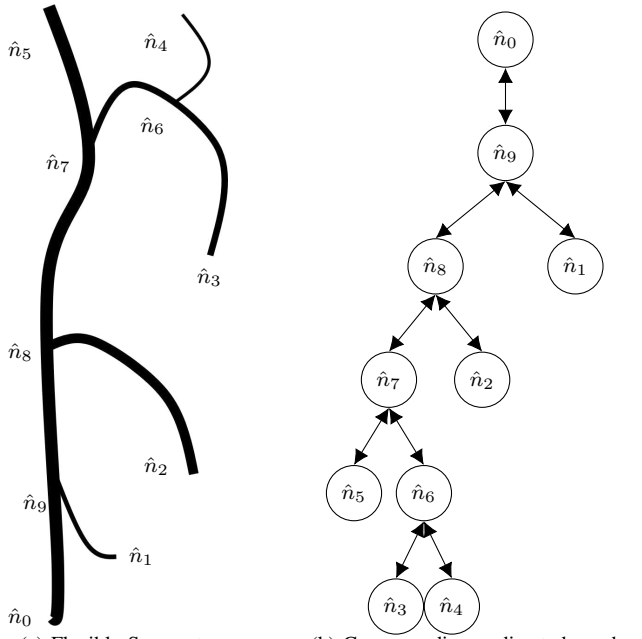
(a) Stretched Segments. (b) Corresponding directed graph.

Figure 2. Segment-wise modelling of BDLOs.

After building the graph according to Fig. 2, it offers a way to determine the leaf nodes of the wire harness. Segment s_i is a leaf segment if the corresponding parent node n_i does not itself have a parent node or its child node n_{i+1} does not have children. If n_0 always represents the starting leaf node with no parent of the wire harness, then all other leaf nodes can be found by evaluating their number of children. In the sketched wire harness of Fig. 2, node n_0, n_5, n_6, n_7, n_8 and n_9 are the leaf nodes. All other nodes contain more than one child.

B. Undirected Graph

Constructing an undirected graph from a 2D or 3D image results in finding the leaf nodes and branch nodes of the BDLO, compare Fig. 3a. Leaf nodes can be found in minimizing the thickness of the binary BDLO represented in Fig. 3a until the segments have a line thickness of one pixel, which is called skeletonization [16]. The used algorithm [16] for skeletonization iteratively deletes the boundary and corner points without deleting multiple at once. Therefore, the algorithm is not sensitive to cables with different radii in pixels but will be significantly slower if the radius in pixels is high. Afterwards the direct neighbors of the pixel can be evaluated. All pixels with just one direct neighbor are leaf nodes. One can then follow the leaf nodes until the pixel has exactly three neighbors, which is determined as a branch node. Two leaf nodes always result in one branch node. The undirected graph is then built combining all the information gathered from the leaf and branch node extraction from the skeletonized image, which results in Fig. 3b. A comparison from the extracted leaf and branch nodes to the topology directly returns information if the view on the BDLO is a partial view, exactly if the extracted number of nodes is less than the expected number of nodes, which results in $j < k$.



(a) Flexible Segments. (b) Corresponding undirected graph.

Figure 3. Building an undirected graph from branch point and leaf point extraction of BDLO images.

For the undirected graph, all estimated leaf nodes $\hat{n}_i, i = 1..(j+1)$ are characterized by having just one neighbor segment $\hat{s}_i = 1$, whereas branch nodes are characterized by having $\hat{s}_i = 3$ segments.

C. Image Processing and Feature Extraction for Topology Matching

The stereo camera returns a 2D image and a 2D disparity map, which indicates the estimated distance for a 3D reconstruction of the 2D image. In the first step, the images have to be segmented from the surrounding scene, such that just the BDLO remains on the images. In this paper, we use background subtraction to obtain the relevant pixel data. As stated in Subsection I-A, DLO image segmentation is also a vital field of research, especially for dynamic environments, where background subtraction is not applicable. If $\mathbf{P} \in \mathbb{R}^{u \times v}$ contains the original image with $u \times v$ pixels of the scene and $\mathbf{P}_B \in \mathbb{R}^{u \times v}$ contains the background of the scene, then the BDLO can be extracted from the original image using a binary mask $\mathbf{M}_B \in \mathbb{N}^{u \times v}$, such that

$$\mathbf{M}_B(i,j) = \begin{cases} 1, & \text{if } |\mathbf{P}_B - \mathbf{P}| < t_B. \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Threshold t_B defines the sensitivity to brightness changes. The binary mask of the used wire harness is shown in Fig. 4.

For leaf and branch node detection, the binary image has to be thinned until the maximum thickness of the BDLO is

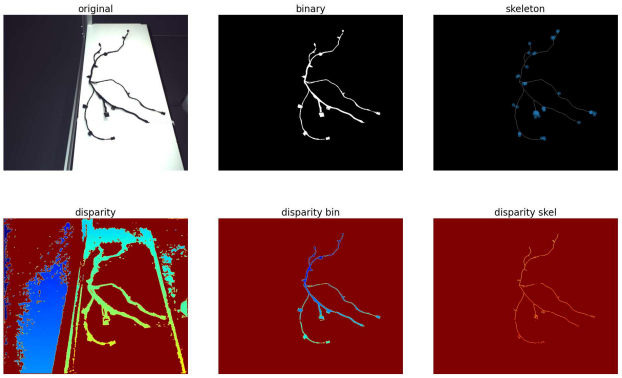


Figure 4. Image processing background subtraction and skeletonization for leaf and branch node extraction. Blue circles indicate possible leaf nodes according to the MST.

one pixel. The thinning was done using the skeletonization algorithm from [16], with the result shown in Fig. 4 on the right. Note that all steps can be repeated for the disparity image, as the disparity image shares the same camera sensor position, such that a 3D reconstruction can be done on reduced data.

After skeletonization, the leaf nodes and branch nodes can be found using a minimum spanning tree (MST). The MST is calculated using the distances of each pixel to each other, such that the distances represent a fully connected spanning tree. Distances are represented in the edges of the tree. Using the MST algorithm, one can find a spanning tree whose sum of edges is minimal, meaning in this case that it is just connected to its closest pixels. This assumes a non-overlapping structure of the BDLO and helps to connect the occluded image parts, as the MST ensures a connection between all pixels of the skeletonized image. The interested reader is referred to [17] for details about the MST algorithm.

For detecting the branch and leaf nodes of the skeleton, we have to count the connected edges in the MST. As depicted in Subsection II-B, a node is a leaf node if $s_i = 1$ and is a branch node if $s_i = 3$. Preprocessing steps assuring long segments deal with corner cases, in which $s_i > 3$ are reduced to $s_i = 3$, in the case of the used BDLO. The resulting leaf node detection is shown with blue circles in Fig. 4.

The skeletonization algorithm creates minor artifacts from the components, which then have to be reduced until the number of leaves equals the number of expected leaves in the topology T. An assumption is that all little leaves created by the components attached to the BDLO are geometrically shorter than the searched leaf nodes. Therefore, 3D reconstruction using the disparity map and using the MST, which still shares the same information about the connections of the MST, allows for estimating the length until a detected leaf node reaches a branch node with edge count three. The distance can be accumulated and afterwards the shortest distance from a leaf node to a branch node can be obliterated until the number of leaf nodes matches the number of expected leaves according to the topology T.

After extracting all nodes from the image, the undirected graph is built according to Subsection II-B. The undirected graph is then used to find all topological matchings between T and the estimated G from the images. All possible solutions $\mathcal{X}$ can be recursively found by Algorithm 1.

Algorithm 1 Building the Solution Space $\mathcal{X}$

Input: Directed graph T and undirected graph G.
Output: Solution Space Vec(Vec($\hat{s}_i$,s_j)).
Initialisation :
empty $\mathcal{X}$
Recursive Matching for all possible leaf segments
for each segment $\hat{s}_i \in G$ **do**
  if (numberOfNeighborSegments($\hat{s}_i$) == 2) **then**
    $s_j \leftarrow s_0$
    recursiveMatching(G, T, $\mathcal{X}$, $\hat{s}_i$, s_j)
  end if
end for
return $\mathcal{X}$

Temporary solutions of correspondences are stored in $\mathbf{X}_{temp}$ in Algorithm 2 and saved into $\mathcal{X}$ when all segments are matched, such that the temporary correspondence equals a valid full correspondence matching.

Algorithm 2 Function for Recursive Matching

Input: G, T, $\mathcal{X}$, $\hat{s}_i$, s_j, optional: $\mathbf{X}_{temp}$.
if length($\mathbf{X}_{temp}$) = number of segments **then**
  $\mathcal{X}$.push_back($\mathbf{X}_{temp}$)
end if
Check if the number of neighbor segments of $\hat{s}_i$ correspond to the required number of neighbor segments of s_j
if matches($\hat{s}_i$,s_j) **then**
  $\mathbf{X}_{temp}$.push_back($\hat{s}_i$, s_j)
else
  return
end if
for each child s_k of s_i **do**
  $s_i \leftarrow s_k$
  recursiveMatching(G, T, $\mathcal{X}$, $\hat{s}_i$, s_j, $\mathbf{X}_{temp}$)
end for
return

The result of the algorithm is shown in Table I and one solution is shown exemplarily in Fig. 5, where the directed graph is generated from the segments s and the corresponding two nodes to one segment of the undirected graph are next to the individual segments.

The solution space in Table I offers a direct interpretation of the topological wire harness design for topological wire harness matching. Segment s_1 is topologically unique, as it can't change within the solution space. This feature allows defining a topological uniqueness score of a wire harness, which can be defined as

$$\zeta = \frac{u_{s,un}}{k}, \tag{3}$$

Table I
SOLUTION SPACE FOR THE EXAMPLE SHOWN IN FIG. 2.

Num	n_0	n_1	n_2	n_3	n_4	n_5	n_6	n_7	n_8	n_9
1	$\hat{n}_2$	$\hat{n}_8$	$\hat{n}_7$	$\hat{n}_6$	$\hat{n}_9$	$\hat{n}_5$	$\hat{n}_3$	$\hat{n}_1$	$\hat{n}_0$	$\hat{n}_4$
2	$\hat{n}_2$	$\hat{n}_8$	$\hat{n}_7$	$\hat{n}_6$	$\hat{n}_9$	$\hat{n}_5$	$\hat{n}_3$	$\hat{n}_1$	$\hat{n}_0$	$\hat{n}_3$
3	$\hat{n}_2$	$\hat{n}_8$	$\hat{n}_7$	$\hat{n}_6$	$\hat{n}_9$	$\hat{n}_5$	$\hat{n}_4$	$\hat{n}_0$	$\hat{n}_1$	$\hat{n}_4$
4	$\hat{n}_2$	$\hat{n}_8$	$\hat{n}_7$	$\hat{n}_6$	$\hat{n}_9$	$\hat{n}_5$	$\hat{n}_4$	$\hat{n}_0$	$\hat{n}_1$	$\hat{n}_3$
5	$\hat{n}_5$	$\hat{n}_7$	$\hat{n}_8$	$\hat{n}_9$	$\hat{n}_6$	$\hat{n}_2$	$\hat{n}_1$	$\hat{n}_3$	$\hat{n}_4$	$\hat{n}_0$
6	$\hat{n}_5$	$\hat{n}_7$	$\hat{n}_8$	$\hat{n}_9$	$\hat{n}_6$	$\hat{n}_2$	$\hat{n}_0$	$\hat{n}_3$	$\hat{n}_4$	$\hat{n}_1$
7	$\hat{n}_5$	$\hat{n}_7$	$\hat{n}_8$	$\hat{n}_9$	$\hat{n}_6$	$\hat{n}_2$	$\hat{n}_1$	$\hat{n}_4$	$\hat{n}_3$	$\hat{n}_0$
8	$\hat{n}_5$	$\hat{n}_7$	$\hat{n}_8$	$\hat{n}_9$	$\hat{n}_6$	$\hat{n}_2$	$\hat{n}_0$	$\hat{n}_4$	$\hat{n}_3$	$\hat{n}_1$

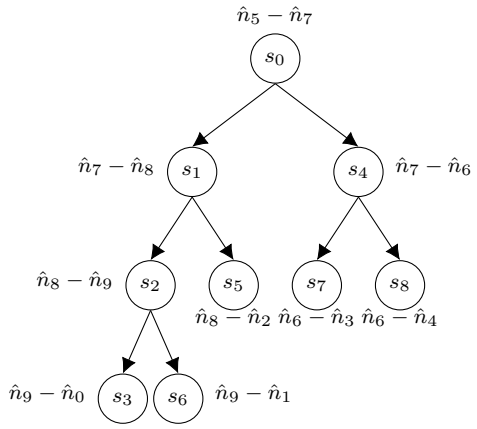


Figure 5. One possible topology matching solution.

if $u_{s,un}$ defines the number of unique segments and k defines the total number of segments. The score $\zeta \in \mathbb{R} : 0 \leq \zeta \leq 1$ defines the topological uniqueness, the higher, the better for topological matching with branch and leaf node extraction. In Fig. 6 different topologies of BDLOs, which all share a different ζ score ranging from 0 to 1, are shown. The ζ score of the BDLO used in Section III is $1/9$, which means that only segment s_1 can be determined by topology matching of branch and leaf nodes. All other segments have to be determined by feature matching, which estimates the likelihood of two nodes belonging to each other according to the solution space $\mathcal{X}$.

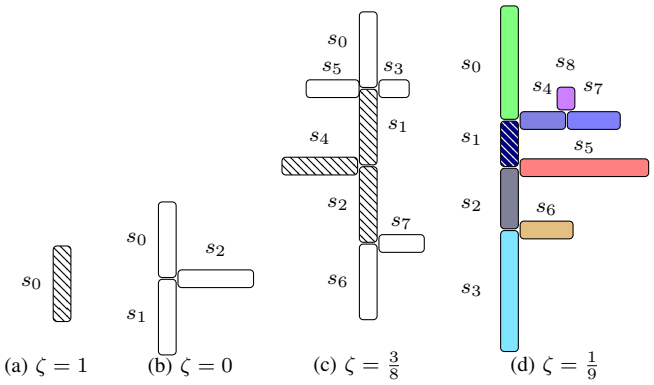


Figure 6. Topological uniqueness (ζ) score for different topologies. Pattern indicate the unique segments of the individual topology.

III. EXPERIMENTS

The camera used in the experiments is a Nerian Scarlet 3D, which performs stereo matching [18] and returns 1216x1024 pixels with an update rate of $20\,\mathrm{s}^{-1}$. The computer used in the experiments has an Intel(R) Xeon(R) W-2145 CPU with a 3.7GHz frequency and 125GB RAM. Algorithms were developed in Python using the packages OpenCV, Scipy, Numpy and Scikit-image.

The evaluation of the topological matching method is tested with 54 samples of random non-overlapping configurations of a BDLO. The data set of the 54 samples is created by manually manipulating the BDLO in various visible different configurations and saving a 2D image, a 2D disparity map, and the intrinsic parameters of the stereo camera. It also contains a background image for background segmentation, as well as the topology of the BDLO by manually measuring the individual stretched segment lengths. The algorithm then computes the correspondence and colors of the individual segments as modeled in Fig. 2a by using the background image, the 2D images, and the intrinsic parameters of the stereo camera.

Generated solutions from the algorithm are evaluated by a segment-wise comparison between the manually measured segment lengths with the estimated segment lengths of the feature matching as depicted in Fig. 7, as well as by manually determining the ground truth correspondence with the color-coded estimated correspondence. Average processing times,

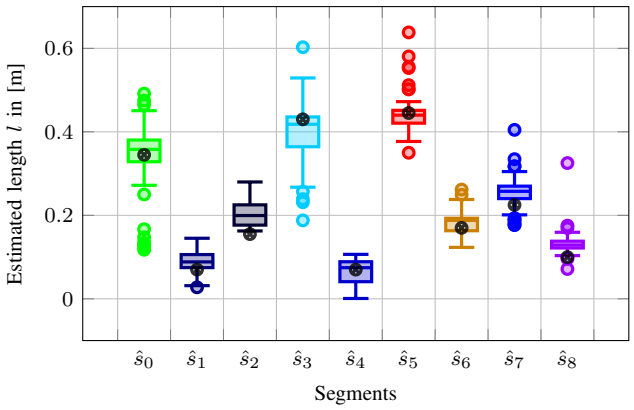


Figure 7. Estimated segment lengths. True values are represented with a black $\otimes$: $l_0 = 0.345m$ $l_1 = 0.07m$ $l_2 = 0.155m$ $l_3 = 0.43m$ $l_4 = 0.07m$ $l_5 = 0.445m$ $l_6 = 0.17m$ $l_7 = 0.225m$ $l_8 = 0.1m$.

the according to average points in the point cloud, and the accuracy over the data set are shown in Table II.

Table II
EVALUATION OF A DATASET WITH 54 SAMPLES

Number of Samples	Average Points in Point Cloud	Average time	Valid Matches	Accuracy
54	2398	3.37s	52	96.3%

Some sample images of the different configurations of the BDLO are shown in Fig. 8. Lengths were estimated using the

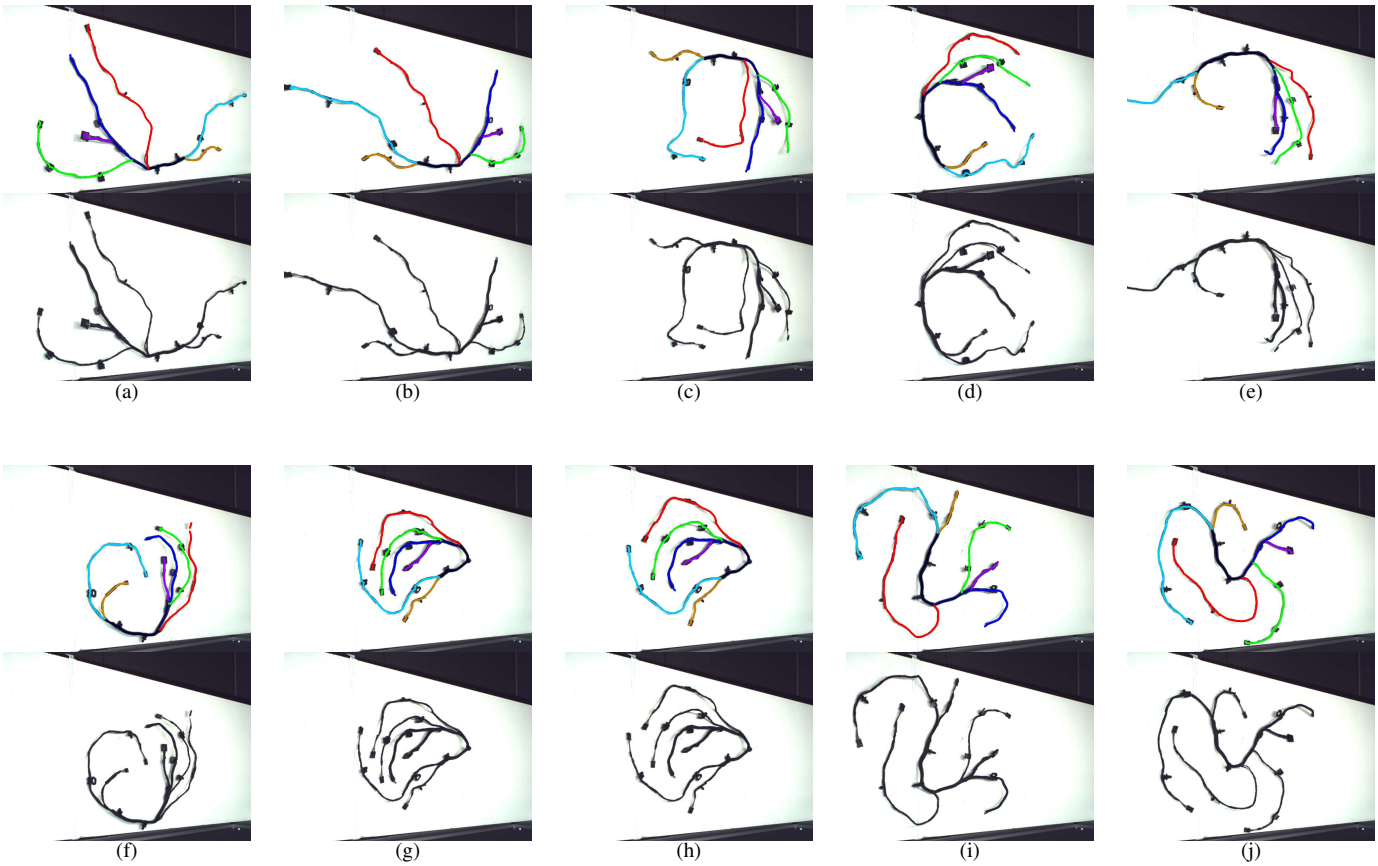


Figure 8. Color-coded matching result after 3D reconstruction. The segment colors are according to the sketches in the modeling section.

farthest point sampling algorithm [19] with 10 points for each segment.

The resulting best match is calculated by

$$f(\mathbf{X}_i) = \sum_{j=1}^{k} |l_j - \mathbf{X}_i \hat{l}_j|. \tag{4}$$

Although length estimation is prone to errors due to outliers having a big impact on the farthest point sampling algorithm, as can be seen in the estimated length's variance in Fig. 7, the effectiveness of the method relies upon reducing the possibilities by building the solution space $\mathcal{X}$ instead of direct branch wise length comparison. Another factor for the high variance is due to segments not being entirely in the 2D camera image, e.g. shown in Fig. 8e for s_3. High topological uniqueness ζ scores would benefit in further reducing the possibilities.

The topological matching had an accuracy of 96.3% in predicting the correspondence between all segments of the directed graph and the extracted undirected graph, as can be seen in Table II. Miss matches occur due to errors in leaf and branch node detection, outlier from the image background segmentation as well as errors during skeletonization.

IV. Conclusions

This paper presents a method for topology matching of a BDLO with camera images for segment-wise correspondence estimation. The proposed estimation method uses segment features to evaluate a matching cost function, which minimizes the feature difference of the modeled topology with the estimated topology. This paper also presents a score for evaluating the topological uniqueness of a BDLO, which is an indicator of how well a BDLO can be topologically matched using the proposed method. The evaluation is performed on a wire harness dataset with 54 unique configurations of a non-overlapping wire harness and an accuracy of 96.3%, taking 3.37s on average to process 1216x1024 RGB-D data and reconstruct the segment-wise skeletonized point cloud from the RGB-D data. Using the method can increase the robustness of non-rigid registration methods by using it as a preprocessing step of the input data and afterwards performing segment-wise non-rigid registration. Furthermore, the presented method could also be used to assist in the manual assembly and installation of wire harnesses, e.g., by combining it with augmented reality to help workers identify branches even if they appear very similar.

The presented implementation is limited to non-overlapping wire scenarios, as it uses MSTs to determine the branch and leaf nodes of the BDLO. Another restriction of the current

implementation is the requirement of a whole view over the BDLO. This is especially challenging for automotive wire harness assembly, as the dimensions of the wire harness might be bigger than the field of view of the camera. The method is also evaluated on one BDLO. Evaluation with further BDLO might show the practicability of the proposed approach but need further investigation.

Future work shall focus on partial matching, as well as combining additional features in the feature matching process, which will increase the applicability and robustness of the implementation.

REFERENCES

[1] H. G. Nguyen, M. Kuhn, and J. Franke, "Manufacturing automation for automotive wiring harnesses," *Procedia CIRP*, vol. 97, pp. 379–384, 2021, 8th CIRP Conference of Assembly Technology and Systems.

[2] X. Yuan and A. Maharjan, "Non-rigid point set registration: recent trends and challenges," *Artificial Intelligence Review*, 2022.

[3] A. B. Conru, "A genetic approach to the cable harness routing problem," 1994.

[4] E. Fedorov and A. Ferenetz, "Computer-aided design of vehicle electrical harnesses," in *2017 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*. Piscataway, NJ: IEEE, 2017, pp. 1–4.

[5] M. Ridel, P. Savi, M. Alberti, J.-P. Parmantier, I. A. Maio, and F. Nardone, "Numerical simulation of aeronautic cable topology and experimental validation," in *International Conference on Electromagnetics in Advanced Applications (ICEAA), 2011*. Piscataway, NJ: IEEE, 2011, pp. 1356–1359.

[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] 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.

[8] M. Wnuk, C. Hinze, M. Zürn, Q. Pan, A. Lechler, and A. Verl, "Tracking branched deformable linear objects with structure preserved registration by branch-wise probability modification," in *2021 27th International Conference on Mechatronics and Machine Vision in Practice (M2VIP)*, 2021, pp. 101–108.

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

[10] X. Jiang, K.-m. Koo, K. Kikuchi, A. Konno, and M. Uchiyama, "Robotized assembly of a wire harness in car production line," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 490–495.

[11] 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 *Computer vision - ACCV 2018*, ser. Lecture Notes in Computer Science, C. V. Jawahar, H. Li, G. Mori, and K. Schindler, Eds. Cham: Springer, 2019, vol. 11362, pp. 662–677.

[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, 182021, pp. 292–298.

[13] 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.

[14] H. Zhou, S. Li, Q. Lu, and J. Qian, "A practical solution to deformable linear object manipulation: A case study on cable harness connection," in *2020 5th International Conference on Advanced Robotics and Mechatronics (ICARM)*. IEEE, 2020, pp. 329–333.

[15] 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*, vol. 81, pp. 387–392, 2019, 52nd CIRP Conference on Manufacturing Systems (CMS), Ljubljana, Slovenia, June 12-14, 2019.

[16] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Commun. ACM*, vol. 27, no. 3, p. 236–239, mar 1984.

[17] R. Graham and P. Hell, "On the history of the minimum spanning tree problem," *Annals of the History of Computing*, vol. 7, no. 1, pp. 43–57, 1985.

[18] K. Schauwecker, "Real-time stereo vision on fpgas with scenescan." [Online]. Available: http://arxiv.org/pdf/1809.07977v1

[19] Y. Eldar, M. Lindenbaum, M. Porat, and Y. Zeevi, "The farthest point strategy for progressive image sampling," *IEEE Transactions on Image Processing*, vol. 6, no. 9, pp. 1305–1315, 1997.