

Learning Configurations of Wires for Real-Time Shape Estimation and Manipulation Planning

Itamar Mishani^a, Avishai Sintov^{b,*}

^aRobotics Institute, Carnegie-Mellon University, 5000 Forbes Ave., Pittsburgh, 12513, PA, USA

^bSchool of Mechanical Engineering, Tel-Aviv University, Haim Levanon St., Tel-Aviv, 6997801, Israel

Abstract

Robotic manipulation of a wire by its ends requires rapid reasoning of its shape in real-time. A recent development of an analytical model has shown that sensing of the force and torque on one end can be used to determine its shape. However, the model relies on assumptions that may not be met in real world wires and do not take into account gravity and non-linearity of the Force/Torque (F/T) sensor. Hence, the model cannot be applied to any wire with accurate shape estimation. In this paper, we explore the learning of a model to estimate the shape of a wire based solely on measurements of F/T states and without any visual perception. Visual perception is only used for off-line data collection. We propose to train a Supervised Autoencoder with convolutional layers that reconstructs the spatial shape of the wire while enforcing the latent space to resemble the space of F/T. Then, the encoder operates as a descriptor of the wire where F/T states can be mapped to its shape. On the other hand, the decoder of the model is the inverse problem where a desired goal shape can be mapped to the required F/T state. With the same collected data, we also learn the mapping from F/T states to grippers poses. Then, a motion planner can plan a path within the F/T space to a goal while avoiding obstacles. We validate the proposed data-based approach on Nitinol and standard electrical wires, and demonstrate the ability to accurately estimate their shapes.

Keywords: Elastic wires, Convolutional Autoencoder, Shape Estimation

1. Introduction

The manipulation of elastic wires is considered a difficult task to operate in industrial environments due to the inability to reason about their shape in real time [1]. Indeed, manipulation tasks in industrial applications such as assembly lines commonly handle only rigid objects. Manipulation of wires, on the other hand, remains in most cases to be operated manually. Reasoning about the shape of a wire along with manipulation capabilities have various applications including cable routing in automotive production lines [2], surgical suturing [3], knot tying [4], hot wire carving [5] and aerial manipulation of cables [6]. Indirectly, the foundations of this work may be applied to cloth folding [7], protein folding [8], tissue manipulation [9, 10], hyper-redundant robots [11] and multi-robot sheet manipulation [12].

The problem of estimating the shape of an elastic rod has been addressed with various approaches. In [13], a simulated discrete elastic rod model is fitted on data obtained from camera images. Similarly, a Fourier series was used to parameterize a cable segmented from an image [14]. In a more recent work, the instabilities of a rod were analyzed by identifying markers with a camera and compared to a numerical simulation [15]. Seminal work by Bretl and McCarthy [16] relied on a description of elastic rods in equilibrium as local solutions to a geometric optimal control problem and showed that the configuration space of the wire is a six-dimensional smooth manifold. The configuration space was also shown to be represented by the Force and Torque (F/T) at the base of the rod. Later, Borum et al. [17] tracked fiducial markers on a planar wire with a camera and fitted them to the model of Bretl and McCarthy to estimate the wire's shape. However, visual perception and image segmenting of thin objects such as a wire in a cluttered environment is a challenging task. Moreover, relying on continuous visual feedback limits the performance of various tasks in which visual uncertainty (e.g., poor

*Corresponding Author.

Email address: imishani@andrew.cmu.edu, sintov1@tauex.tau.ac.il (Avishai Sintov)

lighting or shadows) or occlusion may occur. This may include manipulating the wire within a confined space such as a vehicle frame.

As opposed to vision, using load measurements at the tip of the wire does not require a line-of-sight. The work in [18] demonstrated the estimation of a thin elastic strip using a force sensor and based on a discretized Kirchhoff elastic rod model. The work of Takano et al. [19] relied on a discrete model and measured force and torque to estimate the shape of a thin strip. These approaches are highly dependent on the resolution of the discretization and directly proportional to computation time. Recent work by the authors has proposed a full framework for real-time shape estimation and control of a wire solely using an F/T sensor and without any visual feedback [20]. The work, however, relied on the analytical model devised by Bretl and McCarthy. Since the model does not take into account gravitation, non-linearity of the sensors or other uncertainties, a Neural-Network (NN) was included to calibrate the F/T sensor to map between real sensed loads and theoretical ones defined in the model. However, such process requires solving the inverse problem for each sample where the theoretical load is computed given a measured shape. The computational complexity of such process is high and may take a very long time. Once the F/T sensor has been calibrated, real-time estimation requires repeated solving of a system of ordinary differential equations to find the corresponding shape of the rod. Each solution is computationally expensive and the update frequency remains low [21]. Hence, real-time applications may be limited.

To cope with the computational complexity imposed by wire shape computation, previous work has proposed to pre-compute a roadmap within the free configuration space of the wire as part of a path planning problem [21]. However, the roadmap, acting as a descriptor of the wire, represented only a small subset of possible wire configurations. In this paper, we explore data-based approaches to estimate the shape of the wire given an F/T measurement. A trained Neural-Network (NN) can be an higher-capacity descriptor of the wire enabling rapid estimations of its shape.

In this work, we investigate whether a data-based model can be learned and used for shape estimation of the wire based on simple sensing. We propose to use the Supervised and Convolutional Autoencoder (SnCAE) to learn wire shapes. We rely on the revelation of the work by Bretl and McCarthy [16] where the shape of the wire can be represented in a low-dimension space. An F/T measurement is then assumed to be an encoded state representation of the wire and can poten-

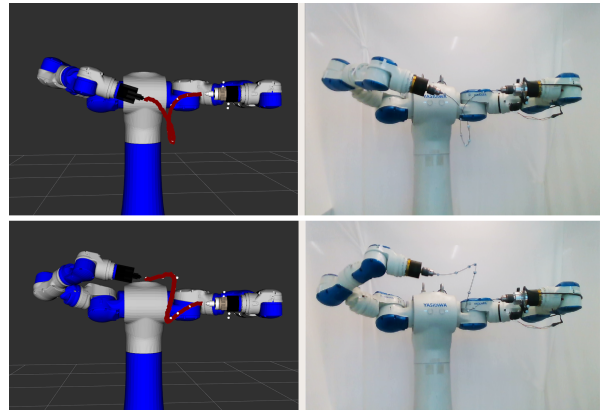


Figure 1: Shape estimation of a Nitinol elastic wire in two different configurations using only a Force/Torque sensor on one gripper. The wire configuration on the right is approximated and visualized in simulation (red curve), and compared to markers tracked (white markers) with a motion capture system.

tially be mapped to its explicit shape. Hence, we train the SnCAE which is a Supervised Autoencoder [22] constructed with convolutional layers. The SnCAE reconstructs the shape of the wire, based on collected data, while also supervising the latent space to match corresponding F/T states. Convolutional layers are included in order to embed the spatial shape of the wire and ease the learning. Once trained, the decoder of the SnCAE is the shape estimator and can rapidly map F/T states to wire shapes. Figure 1 shows an example where a trained decoder is used for real-time estimation of the spatial shape of a wire based on measured F/T and without visual perception. Vision is used solely for collecting training data. We test our approach on Nitinol and standard electrical wires.

In addition to the decoder, the encoder of the SnCAE provides the solution of the inverse problem where an F/T state of the wire can be extracted from a desired shape. Hence, planning in the space of F/T states can be performed while the goal state is extracted using the encoder from a measured shape (e.g., given cable routing channel). Similarly, wire shapes that result in collision with obstacles can be mapped using the encoder to F/T states and avoided in planning. To demonstrate planning with the learned model, the same training data is used to train an NN to map an F/T state to the corresponding pose of one gripper relative to the first. Then, a motion planner is implemented where the encoder identifies goal F/T states and the decoder acts as a collision checker. To summarize, the contributions of this work are as follows:

- We propose a data-based framework to estimate

the shape of a wire based solely on F/T measurements.

- An NN architecture is proposed based on the Supervised Autoencoder. The Autoencoder, composed of an encoder and decoder, is forced to embed the F/T space of the wire in the latent space. The decoder can be used for mapping F/T sensing to the shape of the wire.
- In addition to the decoder, the encoder provides a solution to the inverse problem where a desired shape of the wire is mapped to the required F/T state.
- Convolutional layers are used to embed the spatial shape of the wire in the model.
- Unlike prior work, the proposed model embeds uncertainties such as non-linearity of the F/T sensor, initial wire curvature and non-homogeneity.
- An additional model maps F/T sensing to the pose of the second gripper relative to the base gripper.
- The proposed models are demonstrated in path planning and manipulation of Nitinol and electric wires.
- The work shows that F/T measurements can be mapped to wire shapes even if the wire does not cope with the assumptions made by Bretl and McCarthy [16], e.g., we learn mapping for a non-homogeneous electric wire with initial curvature.

To the best of the authors' knowledge, this is the first attempt to fully describe the shape of a wire using a data-based model and based on F/T measurements. While additional work must be done for generalization, our approach is a first step towards accurate description of wires without dependence on limited analytical models.

2. Related Work

In this section, we survey some topics related to the manipulation of deformable objects and, in particular, elastic wires. The manipulation of deformable objects has been widely researched [23]. Early work used finite element modeling to study the control of static deformations of sheet metal parts handled by two manipulators [24]. In [23], position control was used to manipulate a flexible object with multiple robots while applying vibration suppression at each contact. Different approaches considered the deformable object as a controlled dynamic system [25, 26].

Robotic manipulation of a wire was traditionally considered in the configuration space of its two ends or often of the two grippers grasping them. However, multiple solutions of wire shape exist for a single configuration of the ends. In addition, a representation for the

shape of the wire has infinite dimension. These challenges hinder the motion planning of a wire. Early work on path planning for elastic wires suggested the sampling of gripper displacements and using numerical simulations to approximate their effect on the wire [27, 28]. Later work relaxed gripping points constraints while planning a collision-free path for a sphere around a predefined central grip point [29]. Different approaches use numerical methods to describe the curvature of the deformed wire [30, 31]. However, the above methods are computationally expensive making them hard to perform well. Simplification of the model for the deformed object is another approach where a sequence of rigid masses with springs is used to represent the object [32, 33]. In such approach, the solution is highly sensitive to the approximation which in turn affects the quality of the planning.

In all the cited approaches, a feasible procedure to derive the free configuration space of a wire was not clear at that time. As discussed in previous section, Bretl and McCarthy later showed that the configuration space of the wire, i.e. the set of all equilibrium configurations, is a six-dimensional smooth manifold [16, 34]. They also provided a computational test to determine whether an equilibrium configuration on the manifold is stable or not. This enabled the use of sampling-based planning algorithms [35] to be used in which configurations of the wire could be sampled directly. Further research following the work of Bretl and McCarthy explored additional planning properties in the free configuration space of the wire [36, 37]. An important work has provided the insight that the free configuration space is path-connected and a semi-analytical feasible path can easily be found [38].

While making a breakthrough, the work of Bretl and McCarthy imposed strong assumptions for the wire to be straight in the undeformed state and ignored the effects of gravity. Nevertheless, recent work have shown that adding the Darboux vector to the model can be used to include distributed forces such as gravity and initial wire curvature [39]. The approach was demonstrated on short rubber rods with low gravity influence. In addition, the approach remains to be based on an analytical model with no ability to cope with various model uncertainties such as gripper inaccuracies, intrinsic non-linearity of the sensor and non-homogeneity of the wire.

While less related to the presented work, it is also worth mentioning additional work related to perception and manipulation of linear elastic objects. Recent work demonstrated model-based control for manipulation of ropes on a plane with a single robot arm [40]. Image perception is used for estimating the state of the rope

and the robot can grasp any point along it for a manipulating action. Similar work used synthetic depth images to train a policy for such manipulation on a plane [41]. In [42], a data-driven model was proposed for the dynamics of a rope in various manipulation tasks. The state definition of a rope was degraded to some small set of points on it while not putting state estimation in scope. In a different work, shape estimation of deformable objects was obtained through RGB-D perception in an occluded scene [43].

3. Background

We briefly present the theoretical background from Bretl and McCarthy [16] set to be the baseline to our work. Given a wire of length L , the following model assumes that it is straight in the undeformed configuration with high enough stiffness in which the effects of gravity are neglected. Using $t \in [0, L]$ to denote arc-length along the wire, the position and orientation of the wire at arc-length t are described by a continuous map $\mathbf{q}: [0, L] \rightarrow SE(3)$ given by

$$\mathbf{q}(t) = \begin{bmatrix} R(t) & \mathbf{p}(t) \\ \mathbf{0} & 1 \end{bmatrix}, \quad (1)$$

where $R(t): [0, L] \rightarrow SO(3)$ and $\mathbf{p}(t): [0, L] \rightarrow \mathbb{R}^3$ are curvature and position functions, respectively. Illustration of the wire is seen in Figure 2. According to the Kirchhoff model, a wire is unshearable and inextensible while allowed to bend or twist [44]. These are enforced by requiring \mathbf{q} to satisfy

$$\dot{\mathbf{q}} = \mathbf{q} \begin{pmatrix} \widehat{\mathbf{u}} & \mathbf{e}_1 \\ 0 & 0 \end{pmatrix}, \quad (2)$$

for some function $\mathbf{u}: [0, L] \rightarrow \mathbb{R}^3$, where overdots denote differentiation with respect to t , the map $\widehat{\cdot}: \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ satisfies $a \times b = \widehat{ab}$ for all $a, b \in \mathbb{R}^3$, and $\mathbf{e}_1 = [1 \ 0 \ 0]^T$.

Both ends of the wire are held by two robotic arms with grippers. The position and orientation of each point $\mathbf{q}(t)$ on the wire is represented relative to the *base gripper* at $t = 0$ such that $\mathbf{q}(0) = I$, where $I \in SE(3)$ is the identity matrix. This establishes the initial condition for differential equation (2).

We define the set $\mathcal{A} \subset \mathbb{R}^6$ by

$$\mathcal{A} = \{\mathbf{a} \in \mathbb{R}^6 : (a_2, a_3, a_5, a_6) \neq (0, 0, 0, 0)\} \quad (3)$$

The set \mathcal{A} is simply \mathbb{R}^6 with a two-dimensional plane removed. Each point in \mathcal{A} corresponds to an equilibrium configuration of the wire and a local minimum of the total elastic energy of the wire. Proof for this can

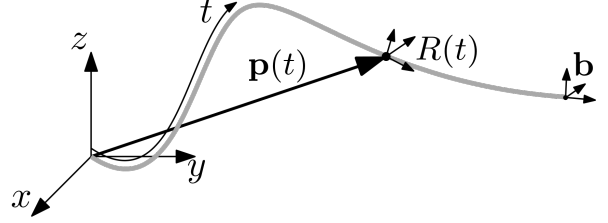


Figure 2: Illustration of an elastic wire (gray). The shape of the wire is given by the position and curvature functions $\mathbf{p}(t) \in \mathbb{R}^3$ and $R(t) \in SO(3)$, respectively, where $t \in [0, L]$ is the arc-length along the wire. The position and orientation of each point along the wire is represented relative to the base gripper at $t = 0$. In addition, the pose of the second gripper at $t = L$ is given by $\mathbf{b} \in \mathcal{B} \subset SE(3)$.

be viewed in Theorem 5 of Bretl and McCarthy [16]. Thus, one can solve the following six ordinary differential equations

$$\begin{aligned} \frac{d\mu_1}{dt} &= \frac{\mu_3\mu_2}{c_3} - \frac{\mu_2\mu_3}{c_2} & \frac{d\mu_4}{dt} &= \frac{\mu_3\mu_5}{c_3} - \frac{\mu_2\mu_6}{c_2} \\ \frac{d\mu_2}{dt} &= \mu_6 + \frac{\mu_1\mu_3}{c_1} - \frac{\mu_3\mu_1}{c_3} & \frac{d\mu_5}{dt} &= \frac{\mu_1\mu_6}{c_1} - \frac{\mu_3\mu_4}{c_3} \\ \frac{d\mu_3}{dt} &= -\mu_5 + \frac{\mu_2\mu_1}{c_2} - \frac{\mu_1\mu_2}{c_1} & \frac{d\mu_6}{dt} &= \frac{\mu_2\mu_4}{c_2} - \frac{\mu_1\mu_5}{c_1} \end{aligned} \quad (4)$$

on the interval $t \in [0, L]$ with the initial condition $\mu(0) = \mathbf{a}$ for $\mathbf{a} \in \mathcal{A}$. In addition, $c_1 > 0$ is the torsional stiffness of the wire and $c_2, c_3 > 0$ are the bending stiffnesses. Furthermore, $u_1: [0, L] \rightarrow \mathbb{R}$ and $u_2, u_3: [0, L] \rightarrow \mathbb{R}$ are the twisting and bending strains along the wire, respectively, such that $\mathbf{u} = (u_1, u_2, u_3)^T$ and $u_i = \mu_i/c_i$ for $i = 1, 2, 3$. Solving (2) with the resulting \mathbf{u} produces an equilibrium shape of the wire, denoted by the pair of functions (\mathbf{q}, \mathbf{u}) . Each (\mathbf{q}, \mathbf{u}) and the corresponding μ are completely defined by the choice of $\mathbf{a} \in \mathcal{A}$. Therefore and in practice, \mathcal{A} serves as the configuration space of the wire. Since we care only about the shape of the wire, we define map $\mathbf{p}(t) = \Phi(\mathbf{a})$ where $\mathbf{p}(t)$ is extracted from \mathbf{q} according to (1). Map Φ is injective, i.e., for each \mathbf{p} there exists a unique $\mathbf{a} \in \mathcal{A}$. In addition, one can define subspace $\mathcal{B} \subset SE(3)$ which is the space of poses of the *second gripper* and is given by $\mathbf{b} = \mathbf{q}(L) \in \mathcal{B}$ where \mathbf{q} relates to some \mathbf{a} .

Function $\mu: [0, L] \rightarrow \mathbb{R}^6$ is interpreted as the vector of internal forces and torques along the wire. Hence, one can describe the force and torque at point t along the wire as

$$\mathbf{f}(t) = (\mu_4(t), \mu_5(t), \mu_6(t))^T \quad (5)$$

$$\boldsymbol{\tau}(t) = (\mu_1(t), \mu_2(t), \mu_3(t))^T, \quad (6)$$

respectively, where $\mu_j(t)$ is the j^{th} component of $\mu(t)$

[16]. Since Φ is injective, any equilibrium configuration $\mathbf{a} = \boldsymbol{\mu}(0)$ is completely defined by the force $\mathbf{f}(0)$ and torque $\boldsymbol{\tau}(0)$ at the base gripper. In other words, by solely measuring the load exerted on the gripper using an F/T sensor, one can directly acquire configuration \mathbf{a} and, using $\Phi(\mathbf{a})$, solve for the shape of the wire \mathbf{p} . In addition, the load measurement can be used to extract the expected pose of the second gripper \mathbf{b} with respect to the base gripper.

4. Method

4.1. Problem Statement

An elastic wire of length L and mechanical coefficients $\mathbf{c} = (c_1, c_2, c_3)^T$ is held by a dual-arm robotic system. Furthermore, a Force/Torque (F/T) sensor is mounted on one arm and measures the load $\bar{\mathbf{a}} \in \mathcal{A}_{FT}$ exerted by the wire where \mathcal{A}_{FT} is the space of measured wire F/T states. Inspired by the analytical model presented in the previous section, we hypothesize that mapping from F/T measurements to wire shape can be learned from collected data while taking uncertainties into account. Therefore, we explore the learning of a discrete map $\Gamma : \mathcal{A}_{FT} \rightarrow \mathbb{R}^3 \times \dots \times \mathbb{R}^3$. Hence, given a measurement $\bar{\mathbf{a}} \in \mathcal{A}_{FT}$ of the F/T sensor, the map will output a set of m points $\Gamma(\bar{\mathbf{a}}) = \{\bar{\mathbf{p}}_1, \dots, \bar{\mathbf{p}}_m\}$ along the wire where $\bar{\mathbf{p}}_j = \mathbf{p}(\frac{j}{m}) \in \mathbb{R}^3$ for $j = 1, \dots, m$.

4.2. Pose estimation using the Analytical Model

As discussed in previous work [20], a measurement $\bar{\mathbf{a}}$ of the F/T sensor cannot directly be applied to extract the shape with model $\Phi(\bar{\mathbf{a}})$. Many uncertainties such as gravitation, gripper inaccuracies, intrinsic non-linearity of the sensor and non-homogeneity of the wire, were not reflected by the assumptions of the model. Hence, the previous work proposed an NN calibration model to map a measured F/T load $\bar{\mathbf{a}}_j$ to the theoretical wire configuration \mathbf{a}_j . To do so, a dataset is collected by manipulating a wire with known length and coefficients \mathbf{c} through various configurations while recording, for each, the F/T measurement $\bar{\mathbf{a}}_j$ and the corresponding set of V marker locations $\mathcal{F}_j = \{\bar{\mathbf{p}}_{j,1}, \dots, \bar{\mathbf{p}}_{j,V}\}$. Point $\bar{\mathbf{p}}_{j,k} \in \mathbb{R}^3$ is the spatial position of marker k relative to the base gripper measured with a motion capture system. For each sample \mathcal{F}_j , we solve the inverse problem $\mathbf{a}_j = \Phi^{-1}(\mathcal{F}_j)$ to compute the theoretical wire configuration \mathbf{a}_j . This was done by solving the following minimization problem

$$\mathbf{a}_j = \arg \min_{\mathbf{a}} \sum_{k=1}^V \|\bar{\mathbf{p}}_{j,k} - \mathbf{x}_k(\mathbf{a})\|^2 \quad (7)$$

where $\mathbf{x}_k \in \mathbb{R}^3$ is the closest point to $\bar{\mathbf{p}}_{j,k}$ on a wire $\mathbf{p} = \Phi(\mathbf{a})$. Problem (7) is non-linear and therefore was solved with a meta-heuristic global optimizer such as Particle Swarm Optimization (PSO) [45]. The calibration model is, then, a NN model ψ trained with dataset $\{(\bar{\mathbf{a}}_j, \mathbf{a}_j)_{j=1}^N\}$ where N is the number of collected samples. Given F/T measurement $\bar{\mathbf{a}}_i$, the shape of the wire is then computed with $\mathbf{p}(t) = \Phi(\mathbf{a}_i)$ for $\mathbf{a}_i = \psi(\bar{\mathbf{a}}_i)$.

Solving inverse problem (7) for each sample is computationally expensive. Furthermore, in order to acquire a fine calibration model, one must collect a large amount of samples and with high variance. Such process may take very long time while insufficient data yields low accuracy as will be demonstrated in the experiments. Even with enough data points, the model may not sufficiently represent the behaviour of the query wire due to various non-modeled properties. In this work, we take a different approach where we directly map F/T measurements to wire shapes and, by that, incorporate all uncertainties in the model. Therefore, explicit knowledge of coefficients \mathbf{c} is not required nor the tedious solution of inverse problem (7).

4.3. Data Collection

Training data is collected by sampling M wire shapes along with their corresponding F/T measurements. Sampling a continuous representation of the wire shape can be done in several ways including shape segmentation from RGB cameras or from a depth camera if the wire is thick enough. In this work, a motion capture system tracks V reflective markers fixed along the wire. It is noted that the motion capture system is used only for data labeling while deployment is done solely based on F/T sensing. Furthermore, while the approach provides a coarse resolution along the wire, the marker positions are acquired with high accuracy. Sample j taken from the system is in the form $(\bar{\mathbf{a}}_j, \mathcal{F}_j)$ where $\mathcal{F}_j = \{\bar{\mathbf{p}}_{j,1}, \dots, \bar{\mathbf{p}}_{j,V}\}$.

To provide a finer representation of the wire given \mathcal{F}_j , we search for a parametric curve function $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{R}^3$ that would represent the wire with higher resolution. In practice, parametric curve

$$\mathbf{f}(t) = (f_x(t), f_y(t), f_z(t))^T \quad (8)$$

could be represented by polynomial functions of degree h such that

$$f_x(t) = \sum_{k=0}^h s_{x,k} t^k, \quad f_y(t) = \sum_{k=0}^h s_{y,k} t^k, \quad f_z(t) = \sum_{k=0}^h s_{z,k} t^k$$

where $s_{j,k}$ are coefficients to be optimized. Curve function $\mathbf{f}(t)$ that best fits points \mathcal{F}_j can be obtained by the

least-squares method and is the solution of

$$\min_v \sum_{k=1}^V \|\bar{\mathbf{p}}_{j,k} - \mathbf{f}(t)\|^2 \quad (9)$$

where v is the vector of all polynomial coefficients in $\mathbf{f}(t)$. Problem (9) can be solved rapidly with parametric curve fitting [46]. In brief, an iterative algorithm associates value t_i to $\bar{\mathbf{p}}_i$ by locally minimizing $\|\bar{\mathbf{p}}_i - \mathbf{f}(t_i)\|^2$ and ensuring that $t_i < t_{i+1}$. With the acquired parametric curve fit, we generate m equally spaced points $\mathcal{P}_j = \{\mathbf{p}_1, \dots, \mathbf{p}_m\}$ along the wire ($m \gg V$). These m points are considered an higher resolution representation of the sampled wire and further used for learning map Γ . Finally, the generated training data consists of M samples in the form $\mathcal{D} = \{(\bar{\mathbf{a}}_i, \mathcal{P}_i)\}_{i=1}^M$. For each $\bar{\mathbf{a}}_i$ sample, we also record the corresponding pose of the second gripper $\bar{\mathbf{b}}_i$. Hence, we have another dataset $\mathcal{V} = \{(\bar{\mathbf{a}}_i, \bar{\mathbf{b}}_i)\}_{i=1}^M$ for motion planning as will be discussed later.

4.4. Learning Model: Supervised and Convolutional Autoencoder

A Fully-Connected NN (FC-NN) model can be directly trained with \mathcal{D} to acquire an approximation of Γ . In such case, the input would be six-dimensional while the output is the flattening of \mathcal{P} to a vector of dimension $3m$. However, such vector representation loses the spatial relationship between coordinates along the wire and may affect accuracy. Alternatively, we propose to incorporate convolutional layers in order to embed spatial computation in the model and allow parameter sharing. On top of that, we train a Supervised Autoencoder (SAE) model [22] to augment the learning and acquire an inverse solution Γ^{-1} along the way. Hence, we further describe the architecture of the proposed Supervised and Convolutional Autoencoder (SnCAE) to learn wire shape representation.

A standard Autoencoder (AE) is a neural-network aimed to find a lower-dimensional embedding of some data, i.e., dimensionality reduction [47]. AE is trained to reconstruct the input at the output through an encoder and a decoder. The encoder is used to identify embedded information in the data and compress it to a latent representation $\mathbf{z} \in \mathbb{R}^d$ where dimension d is lower than the one of the input data. The decoder, on the other hand, reconstructs the original data from the latent representation. AE is normally trained to reconstruct input data X by minimizing the objective function $\|X - X'\|^2$ where X' is the output of the decoder. AE is capable of learning complex non-linear relations where simpler models of dimensionality reduction under-perform.

As mentioned, we preserve the spatial representation of the wire by having the input and reconstructed output to the encoder and decoder, respectively, as $m \times 3$ arrays (each row is a point along the wire). Therefore, the input to the encoder passes through a convolution with an $m_c \times 1$ kernel yielding a convolutional layer of size $m \times 3 \times m_a$. The data then passes through a set of three fully-connected layers of size $3m \cdot m_a \times 1$, $m_b \times 1$ and 6×1 as seen in Figure 3. The latent space is six-dimensional to match the size of the F/T state. The encoder and decoder are mirrored while the output of the decoder goes through a de-convolutional layer of size $m_c \times 1$ yielding an $m \times 3$ array. m_a , m_b and m_c are hyper-parameters to be further optimized. Therefore, the reconstruction loss is given by

$$\mathcal{J}_r = \|\mathcal{P} - \mathcal{P}'\|^2. \quad (10)$$

While the AE is an unsupervised method, SAE is a variation of AE where the model also supervises the representation of the latent space. Based on prior work discussed above, the latent representation of a wire shape is known to be six-dimensional. Hence, we set the latent layer in the AE to be $d = 6$, i.e., $\mathbf{z} \in \mathbb{R}^6$. Furthermore, we add a soft constraint on the latent space to minimize the distance to $\bar{\mathbf{a}} \in \mathcal{A}_{FT}$. Hence, we formulate a latent loss value in the form of

$$\mathcal{J}_a = \|\mathbf{z} - \bar{\mathbf{a}}\|^2. \quad (11)$$

Additionally and for regularization, for each batch of data, we compare the reconstruction of \mathcal{P} with either using \mathbf{z} or $\bar{\mathbf{a}}$ in the decoder. Consequently, we train the SnCAE to minimize a combined loss function

$$\mathcal{J} = \mathcal{J}_{r_z} + \mathcal{J}_{r_a} + w\mathcal{J}_a \quad (12)$$

where \mathcal{J}_{r_z} and \mathcal{J}_{r_a} are the reconstruction loss when applying \mathbf{z} and $\bar{\mathbf{a}}$ to the decoder, respectively. Scalar $w > 0$ is a tunable weight. An illustration of the SnCAE is given in Figure 4. Preliminary analysis has shown that adding \mathcal{J}_{r_a} to the loss improves accuracy by approximately 25%. A trained SnCAE with minimal loss \mathcal{J} can reconstruct wire shape data while giving a physical and practical meaning to the latent space.

The trained SnCAE has two usages as seen in Figure 5. First, the decoder is the approximated mapping Γ which maps a measured $\bar{\mathbf{a}}_i$ to the spatial shape of the wire \mathcal{P}_i . In other words, we can use the decoder as a shape estimator based on F/T sensing at the base gripper. Furthermore, the encoder provides a fast inverse solution instead of problem (7), i.e., $\bar{\mathbf{a}}_i = \Gamma^{-1}(\mathcal{P}_i)$. Therefore, the encoder can be exploited to estimate the required F/T load at the base gripper, i.e., wire state in \mathcal{A}_{FT} , based on a desired shape of the wire.

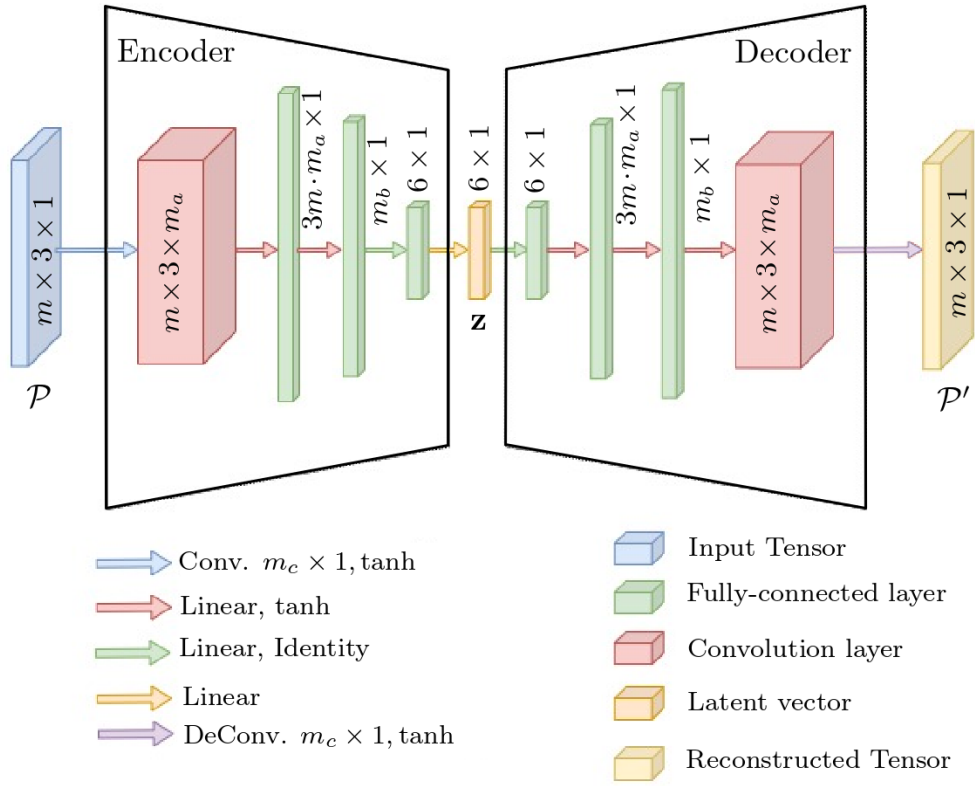


Figure 3: Convolutional layers constructing the encoder and decoder

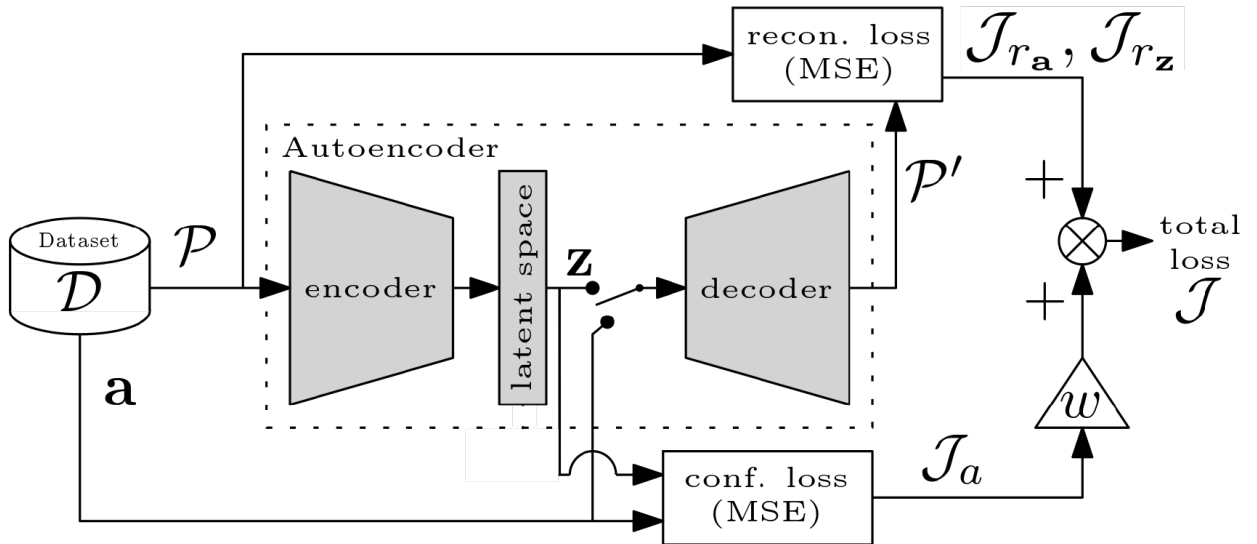


Figure 4: Framework of the Supervised and Convolutional Autoencoder (SnCAE).

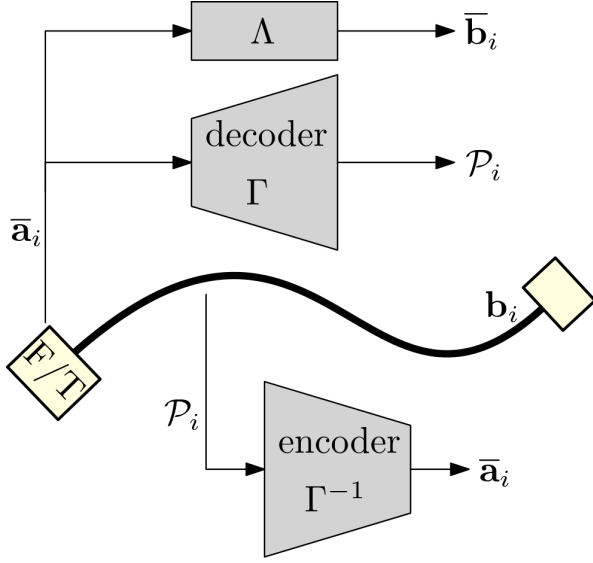


Figure 5: The usages of the trained models given measurements from a grasped wire. The decoder of the SnCAE is used to map an F/T measurement to the shape of the wire, i.e., $\mathcal{P}_i = \Gamma(\bar{\mathbf{a}}_i)$; The encoder of the SnCAE solves the inverse problem and maps a measured shape of the wire to an estimated F/T load at the base gripper, i.e., $\bar{\mathbf{a}}_i = \Gamma^{-1}(\mathcal{P}_i)$; Model Λ approximates the pose of the second gripper given an F/T measurement, i.e., $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$.

We note that an NN architecture similar to the decoder of the SnCAE, termed Decoder CNN (D-CNN), can be trained independently as an alternative to the SnCAE. In D-CNN, model Γ is trained directly without the encoder to map an F/T measurement $\bar{\mathbf{a}}_i$ to wire shape \mathcal{P}_i as seen in Figure 6. Hence, the model is trained to minimize loss

$$\mathcal{J}_D = \|\mathcal{P}_i - \Gamma(\bar{\mathbf{a}}_i)\|^2. \quad (13)$$

However, D-CNN provides only shape estimation while the SnCAE provides both shape estimation and inverse model within the same training. The D-CNN will also be analyzed in the experiments.

4.5. Motion planning over F/T states

A wire configuration $\mathbf{a}_i \in \mathcal{A}$ is given along with its corresponding second gripper pose $\mathbf{b}_i \in \mathcal{B}$. We define an homogeneous transformation matrix $M \in SE(3)$ with $\delta \mathbf{x} \in \mathbb{R}^3$ and exponential coordinates $\mathbf{y} \in \mathbb{R}^3$ and $\delta \theta \in [0, \pi)$ such that

$$M(\delta \mathbf{b}) = \begin{bmatrix} e^{\mathbf{y}\delta\theta} & \delta \mathbf{x} \\ \mathbf{0} & 1 \end{bmatrix} \text{ for } \delta \mathbf{b} = \begin{pmatrix} \mathbf{y}\delta\theta \\ \delta \mathbf{x} \end{pmatrix}. \quad (14)$$

Matrix M is defined to map between two configurations in \mathcal{B} such that

$$\mathbf{b}_{i+1} = \mathbf{b}_i M(\delta \mathbf{b}) \quad (15)$$

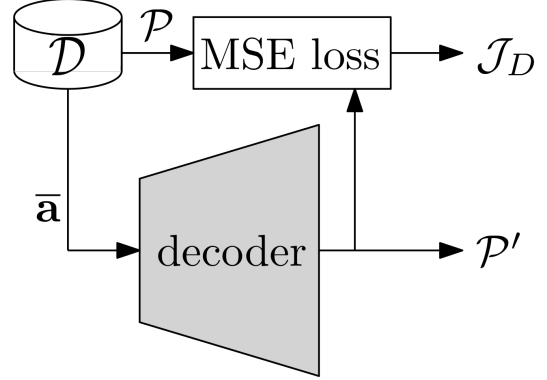


Figure 6: Architecture of the Decoder CNN (D-CNN) used as a baseline comparison.

where perturbation to \mathbf{b}_{i+1} will result in wire configuration \mathbf{a}_{i+1} . From Theorem 7 and equation (37) in [16], we get that

$$\delta \mathbf{b} \approx \mathbf{J}(L)\delta \mathbf{a} \quad (16)$$

where $\delta \mathbf{a} = \mathbf{a}_{i+1} - \mathbf{a}_i$. Matrix $\mathbf{J}(L)$ is the Jacobian relying on the solution of the above ordinary differential equations and is detailed in [20, 21]. Equation (16) states that matrix $\mathbf{J}(L)$ contains information about the relationship between small changes in \mathcal{A} and small changes in \mathcal{B} . Therefore and given a desired \mathbf{a}_{i+1} in the vicinity of \mathbf{a}_i , the required perturbation $\delta \mathbf{a}$ in \mathcal{A} can be obtained. Then, by solving (16), one can use map (15) to compute the required perturbation in \mathcal{B} in order to move a wire from configuration \mathbf{a}_i to \mathbf{a}_{i+1} .

A data-based approach and a learned map Γ do not enable the extraction of an explicit representation of the Jacobian. Furthermore, the non-linearity of the F/T sensor does not guarantee that perturbation $\delta \bar{\mathbf{a}}$ can be mapped to the same $\delta \bar{\mathbf{b}}$ for any $\bar{\mathbf{a}}_i \in \mathcal{A}_{FT}$. Alternatively, we propose to directly learn mapping from $\bar{\mathbf{a}}_i$ to $\bar{\mathbf{b}}_i$, i.e., $\Lambda : \mathcal{A}_{FT} \rightarrow \mathcal{B}$. Once having map Λ , we can extract the required gripper perturbation $\delta \bar{\mathbf{b}}_i$ from $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$ to $\bar{\mathbf{b}}_{i+1} = \Lambda(\bar{\mathbf{a}}_{i+1})$ according to (14)-(15). With dataset \mathcal{V} , an FC-NN is trained to learn map $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$ (Figure 5). It is noted that map Γ can only provide the position of the second gripper while Λ provides the full pose in $SE(3)$.

Matrix $\bar{\mathbf{b}}_i \in SE(3)$ contains a rotation matrix $R_{\bar{\mathbf{b}}_i}$ and the position vector. While the latter is easy to encode, direct encoding of rotation matrices for NN training cannot be done while maintaining orthonormality. Hence, we encode $\bar{\mathbf{b}}_i$ with a nine-dimensional vector by flattening the position vector along with two columns ($\mathbf{v}_1, \mathbf{v}_2$) of the matrix. Reconstruction of the rotation matrix $R_{\bar{\mathbf{b}}_i}$ given output $(\mathbf{v}_1, \mathbf{v}_2)$ from the NN is done

using the Gram–Schmidt process [48] according to

$$\begin{aligned} \mathbf{w}_1 &= \frac{\mathbf{v}_1}{\|\mathbf{v}_1\|} \\ \mathbf{u}_2 &= \mathbf{v}_2 - (\mathbf{w}_1 \cdot \mathbf{v}_2)\mathbf{w}_1 \\ \mathbf{w}_2 &= \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|} \\ R_{\bar{\mathbf{b}}_i} &= [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \mathbf{w}_1 \times \mathbf{w}_2]. \end{aligned}$$

Model Λ could be used to plan motion to a desired goal as follows. Given a wire shape goal \mathcal{P}_g , the F/T state goal is the solution of the inverse problem $\bar{\mathbf{a}}_g = \Gamma^{-1}(\mathcal{P}_g)$ computed with the encoder of the SnCAE. A motion planner would output a continuous path $\gamma : [0, 1] \rightarrow \mathcal{A}_{FT}$ from the current start $\gamma(0) = \bar{\mathbf{a}}_s \in \mathcal{A}_{FT}$ to a desired one $\gamma(1) = \bar{\mathbf{a}}_g \in \mathcal{A}_{FT}$. Furthermore, decoder mapping Γ is used as a collision checker where shapes of candidate F/T states are validated to be collision free. We check wire collision with obstacles or with the robot and wire self-collision. A sampling-based motion planner can then be used for finding a collision-free path [35]. In this work, we employ the asymptotically optimal variant of the Rapidly-exploring Random Tree, i.e., RRT* [49]. The RRT* planner finds a path from $\bar{\mathbf{a}}_s$ to $\bar{\mathbf{a}}_g$ while minimizing path length in \mathcal{A}_{FT} . Once path γ is acquired, a step from $\bar{\mathbf{a}}_i$ to $\bar{\mathbf{a}}_{i+1}$ along the path is translated to perturbation command $\delta\bar{\mathbf{b}}_i$ of the second gripper from $\bar{\mathbf{b}}_i = \Lambda(\bar{\mathbf{a}}_i)$.

5. Experiments

Our experiments are based on a setup comprised of the Yaskawa Motoman SDA10F dual-arm robot seen in Figure 7. A six-axis F/T sensor (Bota SensONE) with a gravity compensation module was mounted on its left arm. Furthermore, chuck grippers on both arms fix the wires. A set of $V = 11$ reflective markers was positioned along the installed wire so that a motion capture system is able to provide ground-truth measurements of its shape in real-time. The system is controlled using the Robot Operating System (ROS) over an Ubuntu machine. Videos of the experiments and demonstration can be seen in the supplementary material.

5.1. Shape estimation analysis

We evaluate shape estimation on two wires seen in Figure 8: a Nitinol wire of 2 mm diameter and 820 mm length, and a standard electrical wire of 3 mm diameter and 500 mm length. The electrical wire does not meet the ground assumptions of [16] and was shown in [20] to yield large approximation errors when using the analytical model. Data was collected as discussed

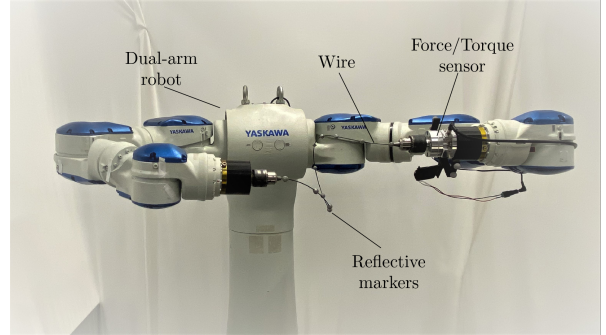


Figure 7: Experimental setup based on a dual-arm robot.

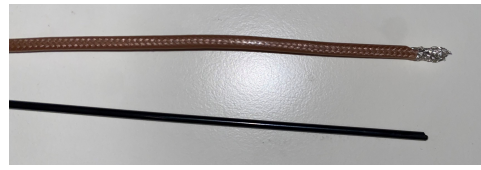


Figure 8: Electrical (top) and Nitinol (bottom) wires used in the experiments.

Table 1: Results for Nitinol and Electric wires shape estimation using various models

	Model	Mean error (mm)
Nitinol	Analytical model [20]	37.14±16.48
	FC-NN	12.16±5.26
	SnCAE (with Decoder)	10.70±0.25
	D-CNN	11.20±0.17
Electric	Analytical model [20]	56.29±19.18
	FC-NN	28.2±6.12
	SnCAE (with Decoder)	22.6±0.57

Table 2: Computation time for Nitinol and Electric wires shape estimation using various models

Model	Comp. time (msec)
Analytical model [20]	72.22±11.40
FC-NN	0.18±0.19
SnCAE (with Decoder)	1.83±5.26
D-CNN	1.72±5.07

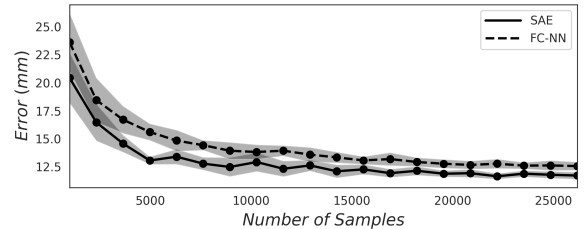


Figure 9: Shape estimation accuracy of the Nitinol wire with regards to the number of collected samples and for FC-NN and SnCAE.

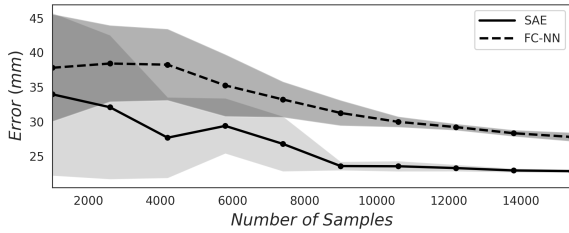


Figure 10: Shape estimation accuracy of the electric wire with regards to the number of collected samples and for FC-NN and SnCAE.

Table 3: Accuracy of inverse solution with the SnCAE encoder

Wire	Force error (N)	Torque error (Nm)
Nitinol	0.34 ± 0.28	0.038 ± 0.032
Electric	0.52 ± 0.11	0.051 ± 0.016

in Section 4.3 while sampling $M_{Nitinol}=26,187$ and $M_{Electric}=15,400$ Nitinol and electric wire shapes, respectively, and their corresponding F/T measurements. For each sample, an $h = 6$ degree polynomial function was fitted and $m = 100$ equally spaced points were generated along it. Training set \mathcal{D} is, therefore, comprised of F/T measurements and their corresponding polynomial approximation of the shape. In addition, test sets were collected independent of the training set and included approximately 1,200 samples. Shape estimation error is defined as the Root-Mean-Square-error (RMSE) between measured marker positions of a test shape and the closest points of the predicted polynomial shape (based on corresponding F/T measurement).

We first analyze the shape estimation of the Nitinol wire. Using dataset \mathcal{D} , we train the proposed SnCAE model along with FC-NN and D-CNN models for comparison. In addition, we also provide results for the analytical model as implemented in [20]. Hence, optimization problem (7) was solved for each sample in \mathcal{D} taking approximately 48 hours in total. With the solutions, we train NN model ψ to map F/T measurements to theoretical wire configurations enabling the solution of $\mathbf{p}(t) = \Phi(\psi(\bar{\mathbf{a}}_t))$. On the other hand, the other NN models were optimized to yield the lowest shape estimation error. The optimal FC-NN model is

Table 4: Roll-out errors along planned paths

Error	Shape (mm)	Force (N)	Torque (Nm)
Goal reach	4.5 ± 2.9	0.23 ± 0.18	0.041 ± 0.027
Path tracking	6.1 ± 3.3	0.36 ± 0.29	0.066 ± 0.055

composed of one hidden layer of 119 nodes, Rectified Linear Unit (ReLU) activation function and a regularizer of 8.8×10^{-7} . The optimal hyper-parameters of the SnCAE model are $m_a = 16$, $m_b = 480$ and $m_c = 10$. As discussed in Section 4.4, the evaluated D-CNN has the same structure of the SnCAE decoder and with similar optimal hyper-parameters.

Table 1 summarizes the shape estimation accuracy results of all methods for the Nitinol and Electric wires. In addition, Table 2 shows the mean computation time of an individual wire shape estimation for both wire types. Computation time was evaluated on an Intel-Core i7-8700 Ubuntu machine with 16GB of RAM. First, the analytical model not only takes a long period of time to process data, it provides inferior results in terms of accuracy. For the model to provide better accuracy as demonstrated in [20], much more data is required for training calibration model ψ with the cost of days more of computation. Also, the average computation time for one shape estimation is rather large, fits to the time reported in [21] and is limited in real-time motion planning. On the other hand, with the same amount of data, a NN can provide a much lower error as the results indicate. Furthermore, SnCAE is shown to provide lower errors in average compared to the FC-NN with a much smaller standard deviation. D-CNN also provides a fairly good accuracy while only providing map Γ without the inverse solution. Computation times of shape estimation for all data-based models are much faster by at least an order of magnitude compared to the analytical model. Note also that with a GPU, the computation time can be significantly reduced. Thus, data-based models are far more suitable for real-time applications.

Figures 9 and 10 show the shape estimation error of the Nitinol and Electrical wires, respectively, for SnCAE and FC-NN with regards to the number of samples in \mathcal{D} . The error for each number of samples was cross validated over 20 sequential data batches taken randomly from the entire training set. While the improvement of SnCAE over FC-NN is marginal for Nitinol, the improvement is much larger for the electrical wire. Overall, the results show that SnCAE outperforms FC-NN while enabling low errors for a relatively small amount of data. The electrical wire is softer than the Nitinol and, therefore, the magnitude of the F/T signals are smaller and more affected by noise making it harder to learn. Nevertheless, the mean error of SnCAE is rather small. In addition, one can settle for half of the data and acquire almost the same accuracy. In a collision checker and motion planning setting, a safety distance would be taken from the wire that is larger than such accuracy. Figure 11 illustrates six Nitinol wire

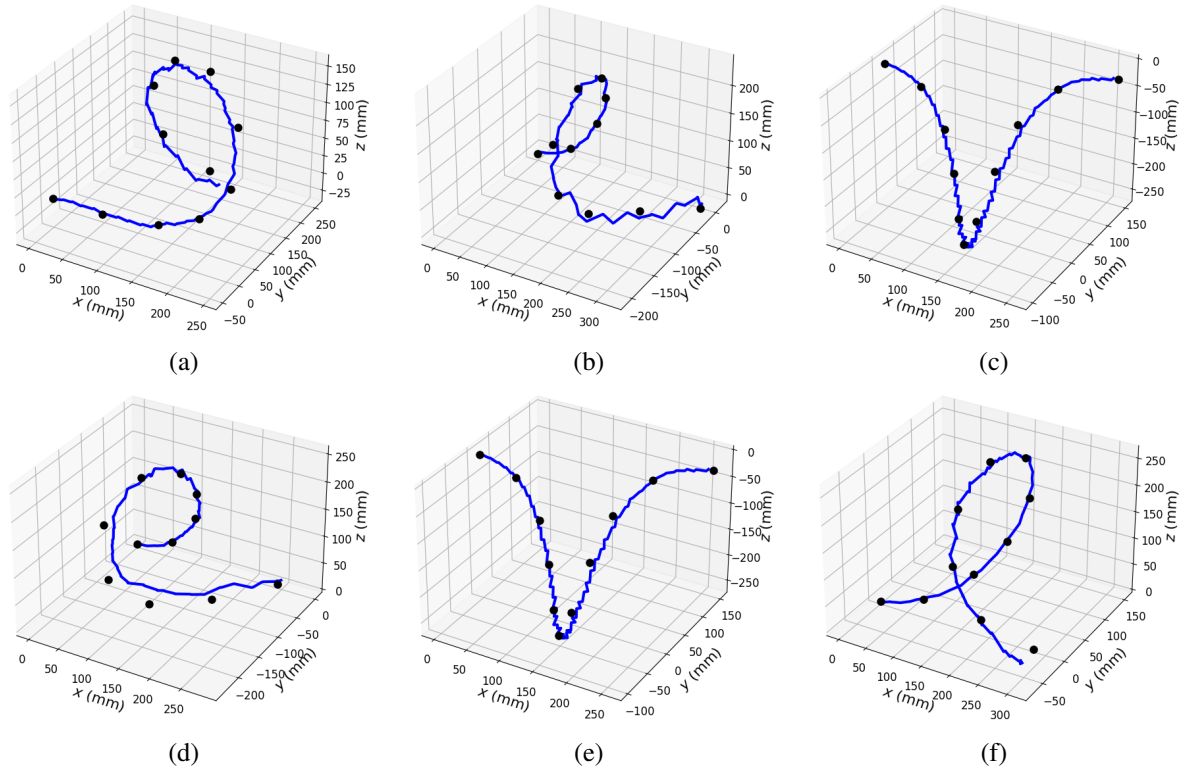


Figure 11: Examples of six Nitinol wire shape estimations. Black markers are the measured ground truth while the blue curve is the shape estimation. The estimations have mean errors of (a) 11.75mm, (b) 9.83mm, (c) 6.54mm, (d) 12.61mm, (e) 8.30mm and (f) 13.32mm.

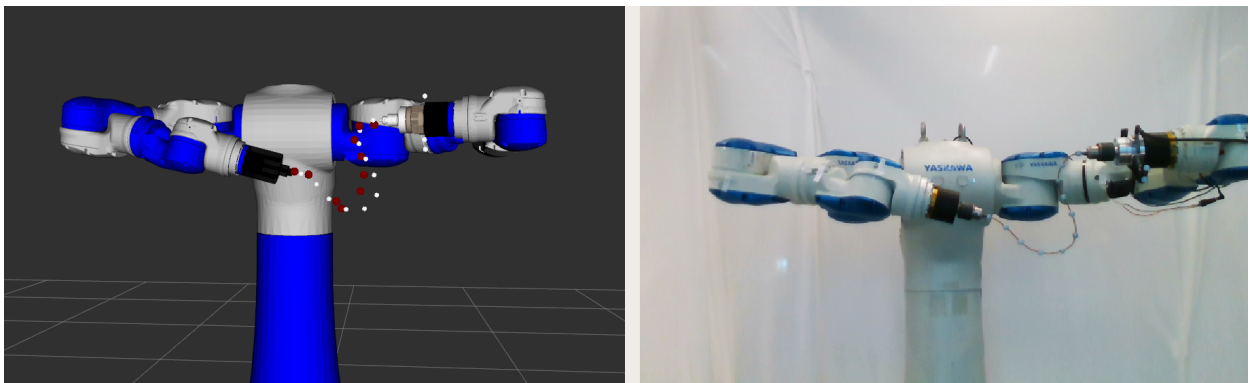


Figure 12: Shape estimation of a standard electrical wire using only a Force/Torque sensor on one gripper. The wire configuration on the right is approximated and visualized in simulation (red markers), and compared to markers (white markers) tracked with a motion capture system.

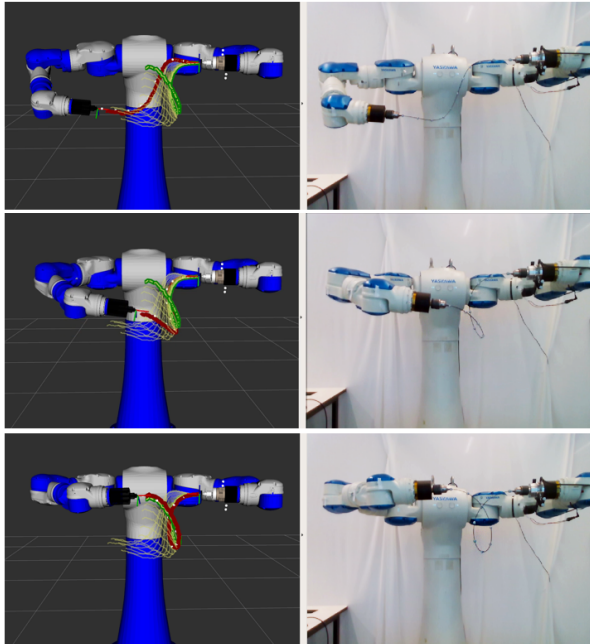


Figure 13: The wire is manipulated along a planned path (yellow curves) in \mathcal{A}_{FT} from (top) start to (bottom) goal. The shape of the wire is estimated in real-time (red curve) using the decoder of the SnCAE and motion of the gripper is determined according to model Λ .

shape estimation. In addition, Figures 1 and 12 show examples of shape estimation with SnCAE in real-time for the Nitinol and electrical wires, respectively.

Table 3 presents the F/T errors of the inverse problem solved with the Encoder of the SnCAE. The encoder is evaluated over the test data mapping measured wire shapes to F/T states. For reference, maximal absolute force and torque measured with the Nitinol wire are 6.27 N and 1.06 Nm, respectively, and for the electrical wire, 3.10 N and 0.3 Nm. The results, therefore, show good F/T estimation allowing mapping desired shapes to F/T states for motion planning, as discussed in the next section.

5.2. Path planning

We conduct an experiment in which the robot must manipulate the Nitinol wire to some desired shape. Map Λ is implemented by training a standard fully-connected NN as discussed in Section 4.5. Furthermore, goal shapes \mathcal{P}_g are randomly sampled by manually placing the cable without the robot along desired shapes and recording using the motion capture system. A goal state is computed with the encoder according to $\bar{\mathbf{a}}_g = \Gamma^{-1}(\mathcal{P}_g)$. An RRT* is then implemented to plan

in \mathcal{A}_{FT} while using Γ for collision checking and Λ for moving the robot along the path.

We analyze roll-outs along ten planned paths to various goals. At each trial, we plan motion from the current shape to a chosen goal shape. Once planned, the path is rolled-out in open-loop by exerting the computed sequence of gripper poses. Table 4 summarizes the results for roll-out accuracy both for tracking the path and reaching the desired goal. Note that a shape tracking error refers to the RMSE of the markers relative to the planned shapes of the corresponding steps along the path. Then, the results show high accuracy tracking and goal reaching. A path was planned with the accuracy of the SnCAE decoder. However, roll-outs are dependent on the accuracy of the corresponding gripper pose $\bar{\mathbf{b}}$ acquired by learned model Λ . Hence, the accurate model Λ enabled the good tracking of paths without regards to the accuracy of the decoder that planned them. Figure 13 shows snapshots of one roll-out while the corresponding path tracking in \mathcal{A}_{FT} is illustrated in Figure 14. Another roll-out example is seen in Figure 15. Tracking is seen to be accurate both in shapes and in \mathcal{A}_{FT} .

Figure 16 shows a demonstration of planning a path in the presence of a cylindrical obstacle. The location of the obstacle was detected with markers and a motion capture system. Motion to the same goal was conducted five times while starting from different states. All roll-outs were successful and the wire did not collide.

6. Conclusion

We have explored the learning of wire shapes based on non-visual perception. While analytical models are commonly used, we have proposed training a NN model based on F/T measurements exerted on a robot arm by a wire. An autoencoder based model, termed SnCAE, was presented where convolutional layers were used to maintain the spatial shape of the wire and the latent space was forced to resemble the space of F/T measurements. Then, the trained encoder and decoder are used for mapping wire shape to F/T state and vice versa, respectively. In general, the evaluated data-based models (decoder of SnCAE and FC-NN) outperformed the analytical approach proposed in prior work for shape estimation. While the SnCAE gained moderate accuracy improvement over the FC-NN for the two cables, it has provided an added value with the encoder. The encoder was shown to be able to solve the inverse problem and identify goal F/T states from given shapes. Furthermore, the results show that sufficient accuracy can be achieved with a relatively small amount of samples.

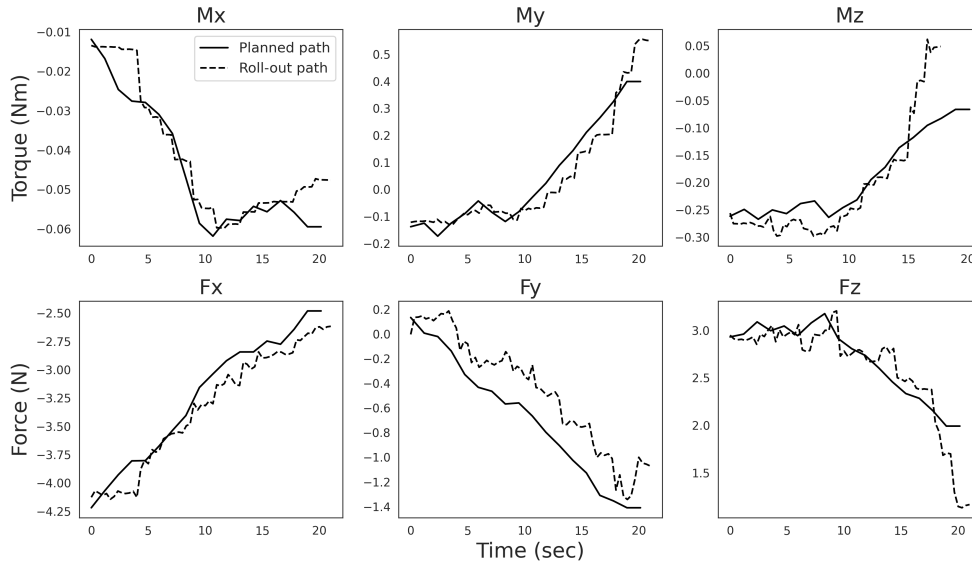


Figure 14: Roll-out of a planned path in \mathcal{A}_{FT} corresponding to the motion in Figure 13.

Along with a Nitinol wire, we have demonstrated, for the first time, the ability to predict the shape of an electrical wire with an accuracy that is feasible for real applications. With another NN model trained with the same collected data, we can map F/T states to desired gripper poses. Hence, a motion planner was implemented to plan and roll-out collision-free paths. A set of experiments was shown to validate pose estimation and planning accuracy. Overall, we have shown the ability to learn an accurate mapping from measured F/T load to the shape of the wire and vice versa.

Our proposed method indeed provides a model for the trained wire. The model cannot be applied, for instance, to a wire of different length. Therefore, future work may involve the generalization of a NN model to wires of various lengths, materials and stiffnesses. In addition, some data augmentation can be performed to reduce the number of real samples required. Alternatively, data collected from a physics engine along with domain randomization may provide sufficient generalization to various wires. Advanced models such as the U-Net[50] can also be integrated in order to augment performance.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] R. Levien, The elastica: a mathematical history, Tech. rep., EECS Department, University of California, Berkeley (Aug 2008).
- [2] X. Jiang, K.-M. Koo, K. Kikuchi, A. Konno, M. Uchiyama, Robotized assembly of a wire harness in a car production line, *Advanced Robotics* 25 (3-4) (2011) 473–489. doi:10.1163/016918610X551782.
- [3] R. C. Jackson, M. C. Çavuşoğlu, Needle path planning for autonomous robotic surgical suturing, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 1669–1675. doi:10.1109/ICRA.2013.6630794.
- [4] J. Schulman, J. Ho, C. Lee, P. Abbeel, Learning from demonstrations through the use of non-rigid registration, in: M. Inaba, P. Corke (Eds.), *Robotics Research: The 16th International Symposium ISRR*, Springer International Publishing, 2016, pp. 339–354. doi:10.1007/978-3-319-28872-7_20.
- [5] S. Duenser, R. Poranne, B. Thomaszewski, S. Coros, Robocut: Hot-wire cutting with robot-controlled flexible rods, *ACM Transactions on Graphics* 39 (4) (2020). doi:10.1145/3386569.3392465.
- [6] F. Augugliaro, E. Zarfati, A. Mirjan, R. D’Andrea, Knot-tying with flying machines for aerial construction, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5917–5922. doi:10.1109/IROS.2015.7354218.
- [7] Y. Avigal, L. Berscheid, T. Asfour, T. Kröger, K. Goldberg, Speedfolding: Learning efficient bimanual folding of garments, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 1–8. doi:10.1109/IROS47612.2022.9981402.
- [8] O. Chang, F. A. Gonzales-Zubiarte, L. Zhinin-Vera, R. Valencia-Ramos, I. Pineda, A. Diaz-Barrios, A protein folding robot driven by a self-taught agent, *Biosystems* 201 (2021) 104315. doi:10.1016/j.biosystems.2020.104315.
- [9] C. Shin, P. W. Ferguson, S. A. Pedram, J. Ma, E. P. Dutton, J. Rosen, Autonomous tissue manipulation via surgical robot using learning based model predictive control, in: *IEEE Interna-*

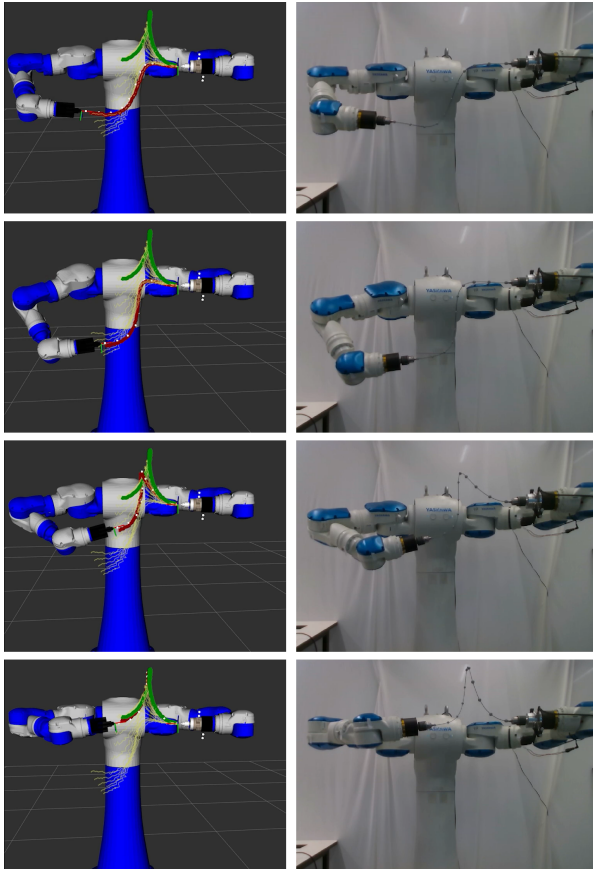


Figure 15: Another example of the wire manipulated along a planned path (yellow curves) in \mathcal{A}_{FT} from (top) start to (bottom) goal. The shape of the wire is estimated in real-time (red curve) using the decoder of the SnCAE and motion of the gripper is determined according to model Λ .

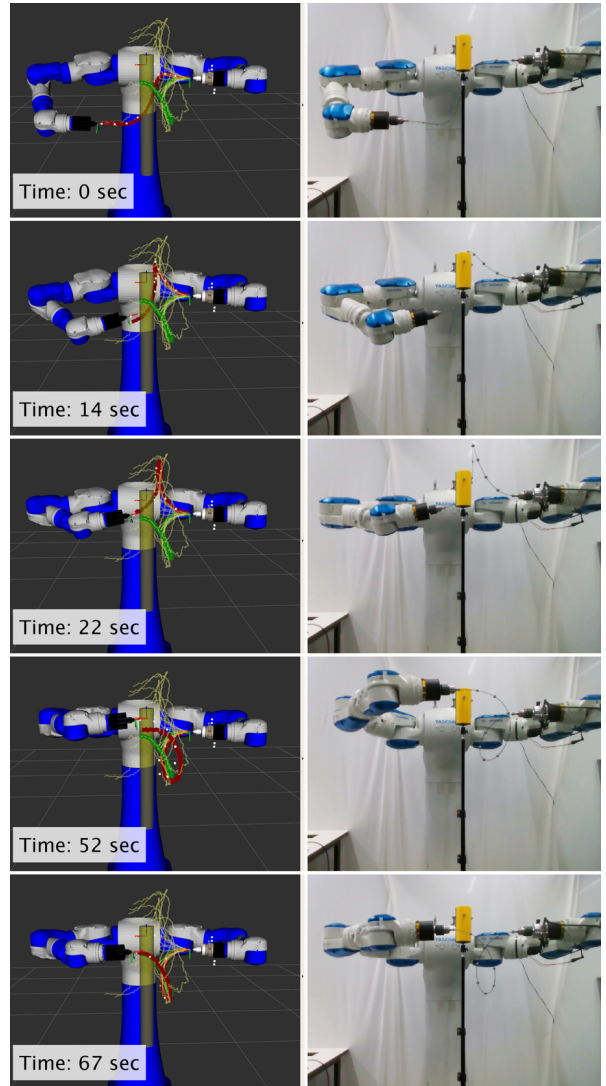


Figure 16: Roll-out of a planned path (yellow curves) across an obstacle (yellow cylinder).

- tional Conference on Robotics and Automation (ICRA), 2019, pp. 3875–3881. doi:10.1109/ICRA.2019.8794159.
- [10] M. Wnuk, F. Jaensch, D. A. Tomzick, Z. Chen, J. Terfurth, S. Kandasamy, J. Shahabi, A. Garrett, M. H. Mahmoudinezhad, A. Csiszar, W. L. Xu, O. Röhrle, A. Verl, Challenges in Robotic Soft Tissue Manipulation—Problem Identification Based on an Interdisciplinary Case Study of a Teleoperated Drawing Robot in Practice, Springer International Publishing, Cham, 2021, pp. 245–262. doi:10.1007/978-3-030-43703-9_20.
- [11] G. Qin, A. Ji, Y. Cheng, W. Zhao, H. Pan, S. Shi, Y. Song, A snake-inspired layer-driven continuum robot, *Soft Robotics* 9 (4) (2022) 788–797. doi:10.1089/soro.2020.0165.
- [12] J. Hu, W. Liu, H. Zhang, J. Yi, Z. Xiong, Multi-robot object transport motion planning with a deformable sheet, *IEEE Robotics and Automation Letters* 7 (4) (2022) 9350–9357. doi:10.1109/LRA.2022.3191190.
- [13] S. Javdani, S. Tandon, J. Tang, J. F. O’Brien, P. Abbeel,

- Modeling and perception of deformable one-dimensional objects, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1607–1614. doi:10.1109/ICRA.2011.5980431.
- [14] J. Zhu, B. Navarro, P. Fraisse, A. Crosnier, A. Cherubini, Dual-arm robotic manipulation of flexible cables, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 479–484. doi:10.1109/IROS.2018.8593780.
- [15] D. Tong, A. Borum, M. K. Jawed, Automated stability testing of elastic rods with helical centerlines using a robotic system, *IEEE Robotics and Automation Letters* 7 (2) (2022) 1126–1133. doi:10.1109/LRA.2021.3138532.
- [16] T. Bretl, Z. McCarthy, Quasi-static manipulation of a kirchhoff elastic rod based on a geometric analysis of equilibrium configurations, *International Journal of Robotics Research* 33 (1) (2014) 48–68. doi:10.1177/0278364912473169.
- [17] A. Borum, D. Matthews, T. Bretl, State estimation and tracking of deforming planar elastic rods, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 4127–4132. doi:10.1109/ICRA.2014.6907459.
- [18] N. Nakagawa, H. Mochiyama, Real-time shape estimation of an elastic rod using a robot manipulator equipped with a sense of force, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 8067–8073. doi:10.1109/IROS.2018.8593946.
- [19] R. Takano, H. Mochiyama, N. Takesue, Real-time shape estimation of kirchhoff elastic rod based on force/torque sensor, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 2508–2515. doi:10.1109/ICRA.2017.7989292.
- [20] I. Mishani, A. Sintov, Real-time non-visual shape estimation and robotic dual-arm manipulation control of an elastic wire, *IEEE Robotics and Automation Letters* 7 (1) (2022) 422–429. doi:10.1109/LRA.2021.3128707.
- [21] A. Sintov, S. Macenski, A. Borum, T. Bretl, Motion planning for dual-arm manipulation of elastic rods, *IEEE Robotics and Automation Letters* 5 (4) (2020) 6065–6072. doi:10.1109/LRA.2020.3011352.
- [22] L. Le, A. Patterson, M. White, Supervised autoencoders: Improving generalization performance with unsupervised regularizers, in: S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, Vol. 31, 2018.
- [23] D. Sun, J. K. Mills, Y. Liu, Position control of robot manipulators manipulating a flexible payload, *International Journal of Robotics Research* 18 (3) (1999) 319–332. doi:10.1177/02783649922066231.
- [24] K. Kosuge, H. Yoshida, T. Fukuda, M. Sakai, K. Kanitani, Manipulation of a flexible object by dual manipulators, in: *IEEE International Conference on Robotics and Automation (ICRA)*, Vol. 1, 1995, pp. 318–323. doi:10.1109/ROBOT.1995.525304.
- [25] H. G. Tanner, Mobile manipulation of flexible objects under deformation constraints, *IEEE Transactions on Robotics* 22 (2006) 179–184. doi:10.1109/TRO.2005.861452.
- [26] D. Navarro-Alarcon, H. M. Yip, Z. Wang, Y. H. Liu, F. Zhong, T. Zhang, P. Li, Automatic 3-d manipulation of soft objects by robotic arms with an adaptive deformation model, *IEEE Transactions on Robotics* 32 (2) (2016) 429–441. doi:10.1109/TRO.2016.2533639.
- [27] F. Lamiroux, L. E. Kavraki, Planning paths for elastic objects under manipulation constraints, *International Journal of Robotics Research* 20 (2001) 188–208. doi:10.1177/02783640122067354.
- [28] M. Moll, L. E. Kavraki, Path planning for deformable linear objects, *IEEE Transaction on Robotics* 22 (4) (2006) 625–636. doi:10.1109/TRO.2006.878933.
- [29] T. Hermansson, R. Bohlin, J. S. Carlson, R. Söderberg, Automatic assembly path planning for wiring harness installations, *Journal of Manufacturing Systems* 32 (3) (2013) 417 – 422. doi:10.1016/j.jmsy.2013.04.006.
- [30] Y. Asano, H. Wakamatsu, E. Morinaga, E. Arai, S. Hirai, Deformation path planning for manipulation of flexible circuit boards, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 5386–5391. doi:10.1109/IROS.2010.5651132.
- [31] H. Wakamatsu, K. Takahashi, S. Hirai, Static modeling of linear object deformation based on differential geometry, in: *International Journal of Robotics Research*, Vol. 23, 2004, pp. 293–311. doi:10.1177/0278364904041882.
- [32] R. Gayle, P. Segars, M. Lin, D. Manocha, Path planning for deformable robots in complex environments, Vol. 1, MIT Press Journals, 2005, pp. 225–232.
- [33] I. Kabul, R. Gayle, M. C. Lin, Cable route planning in complex environments using constrained sampling, in: *ACM Symposium on Solid and Physical Modeling*, NY, 2007, pp. 395–402. doi:10.1145/1236246.1236303.
- [34] T. Bretl, Z. McCarthy, Equilibrium configurations of a kirchhoff elastic rod under quasi-static manipulation, in: E. Frazzoli, T. Lozano-Perez, N. Roy, D. Rus (Eds.), *Algorithmic Foundations of Robotics X: Proceedings of the Tenth Workshop on the Algorithmic Foundations of Robotics*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 71–87. doi:10.1007/978-3-642-36279-8_5.
- [35] S. M. LaValle, *Planning Algorithms*, Cambridge University Press, 2006.
- [36] O. Roussel, A. Borum, M. Taïx, T. Bretl, Manipulation planning with contacts for an extensible elastic rod by sampling on the submanifold of static equilibrium configurations, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3116–3121. doi:10.1109/ICRA.2015.7139627.
- [37] O. Roussel, M. Taïx, T. Bretl, Efficient motion planning for quasi-static elastic rods using geometry neighborhood approximation, in: *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, 2014, pp. 1024–1029. doi:10.1109/AIM.2014.6878215.
- [38] A. Borum, T. Bretl, The free configuration space of a kirchhoff elastic rod is path-connected, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2958–2964. doi:10.1109/ICRA.2015.7139604.
- [39] S. Wu, J. Zhang, D. Wu, Equilibrium manipulation planning for a soft elastic rod considering an external distributed force and intrinsic curvature, *IEEE Robotics and Automation Letters* 7 (4) (2022) 11442–11449. doi:10.1109/LRA.2022.3199823.
- [40] M. Yan, Y. Zhu, N. Jin, J. Bohg, Self-supervised learning of state estimation for manipulating deformable linear objects, *IEEE Robotics and Automation Letters* 5 (2) (2020) 2372–2379. doi:10.1109/LRA.2020.2969931.
- [41] P. Sundaresan, J. Grannen, B. Thananjeyan, A. Balakrishna, M. Laskey, K. Stone, J. Gonzalez, K. Goldberg, Learning rope manipulation policies using dense object descriptors trained on synthetic depth data, in: *IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 9411–9418. doi:10.1109/ICRA40945.2020.9197121.
- [42] P. Mitrano, D. McConachie, D. Berenson, Learning where to trust unreliable models in an unstructured world for deformable object manipulation, *Science Robotics* 6 (54) (2021). doi:10.1126/scirobotics.abd8170.
- [43] Y. Wang, D. Mcconachie, D. Berenson, Tracking partially-occluded deformable objects while enforcing geometric constraints, in: *IEEE International Conference on*

- Robotics and Automation (ICRA), 2021, pp. 14199–14205. doi:10.1109/ICRA48506.2021.9561012.
- [44] S. Antman, *Nonlinear Problems of Elasticity*, Springer, 2005. doi:10.1007/0-387-27649-1.
- [45] J. Kennedy, R. Eberhart, Particle swarm optimization, in: *International Conference on Neural Networks*, Vol. 4, 1995, pp. 1942–1948. doi:10.1109/ICNN.1995.488968.
- [46] M. Grossman, Parametric curve fitting, *The Computer Journal* 14 (2) (1971) 169–172.
- [47] J. Zhai, S. Zhang, J. Chen, Q. He, Autoencoder and its various variants, in: *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2018, pp. 415–419. doi:10.1109/SMC.2018.00080.
- [48] Y. Zhou, C. Barnes, J. Lu, J. Yang, H. Li, On the continuity of rotation representations in neural networks, in: *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5738–5746. doi:10.1109/CVPR.2019.00589.
- [49] S. Karaman, E. Frazzoli, Sampling-based algorithms for optimal motion planning, *International Journal of Robotics Research* 30 (7) (2011) 846–894. doi:10.1177/0278364911406761.
- [50] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241. doi:10.1007/978-3-319-24574-4_28.