2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Development and Evaluation of an Intuitive Flexible Interface for Teleoperating Soft Growing Robots

Haitham El-Hussieny[1,4], Usman Mehmood[1], Zain Mehdi[1], Sang-Goo Jeong[1],
Muhammad Usman[1], Elliot W. Hawkes[2], Allison M. Okamura[3], and Jee-Hwan Ryu[1]

***Abstract*— Mobility by growth is a new paradigm in robotic systems design and their applications in the real world. Soft, tip-extending, or "growing," robots have potential applications including inspection and navigation in disaster scenarios. However, due to their growing capability, such robots create unique challenges for intuitive human control. In this paper, a new flexible interface is proposed to intuitively map human bending commands into movements of the growing robot while providing shape information of the robot in order to improve situational awareness. Several command mappings are proposed, and a subjective study was conducted to assess the intuitiveness of the developed interface and mappings compared with other commercially available interfaces. The interfaces were evaluated using four metrics in two virtual task scenarios. The proposed interface with shape mapping performed better than the other interfaces, especially when the vine robot rolls over unintentionally during complex tasks.**

I. Introduction

Recent decades have witnessed a wide range of applications of continuum robots in congested, uncertain, and hazardous environments [1]. Inspired by the incredible capabilities exhibited by biology, such as elephant trunks, snakes, and octopus tentacles, continuum robots can bend their backbones at many points along their length to overcome the locomotion limitations of rigid robots [2]. However, there are still many challenging situations in which continuum robots face difficulties in navigation due to sticky, slippery, and distal environments [3].

Taking inspiration from plants, an interesting approach to mobility has been recently proposed. Vine-like growing robots, which emulate locomotion by growing, have proved superb performance towards tackling inspection and rescue purposes in the mentioned challenging environments, e.g. [4], [5], [6]. The large length-to-diameter ratios, continuously bending structures, and the growing capability they possess are the key features that allow vine-like robots to penetrate into challenging environments.

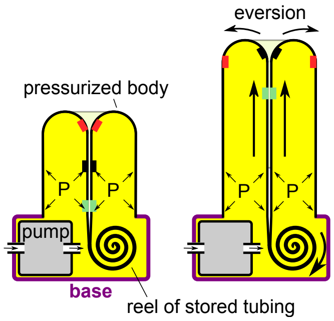


Fig. 1: A schematic of the vine-like tip-extending, or "growing," soft robot. Pneumatic pressure is used to extend the body from the tip by everting new membrane material that is stored in the base. Modified from [4].

Our work is directly inspired by a soft growing robot that can adapt to its surroundings while moving through obstacles in a robust fashion [4]. Its body is made from a thin polyethylene that is folded into itself inside-out as depicted in Figure 1. The robot is driven by air pressure to grow by lengthening at its tip; new material is fed through the body of the robot and allows the tip to travel orders of magnitude further than other vine-like robots. This method of extension reduces the effect of friction with the robot's surroundings, enabling it to move in challenging situations, like sticky and tightly constrained environments compared to other growing robots [7], [8], [9]. A preliminary model for the vine robot is described in [10]. It largely draws from rate-based extension models that describe apical extension in growing cells.

Due to the unstructured nature of the environments that vine robots will have to navigate in search and rescue applications, a human-in-the-loop teleoperation scheme is proposed [11]. One of the key factors that could affect the performance of teleoperation is the design and the mapping strategies used in the human input interface [12]. However, thus far, commercially available input interfaces have been used to teleoperate continuum robots without considering unique kinematic structure and locomotion of them. For instance, a Phantom Omni featuring a pen-like stylus was used in [13], [14], [15], while a 6-DoF rigid manipulator was used in [16] to map both planar and spatial motions to a continuum robot. In [17], a graphical 3D continuum model was used to manipulate the slave robot, and in [18] a joystick input interface was incorporated to map its angle magnitude into the desired curvature of the robot.

This research was partially supported by the project "Toward the Next Generation of Robotic Humanitarian Assistance and Disaster Relief: Fundamental Enabling Technologies (10069072)" and by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP)(No. NRF- 2016R1E1A1A02921594).

[1] School of Mechanical Engineering, Korea University of Technology and Education, Cheonan-si, Republic of Korea `zainmehdi@kut.ac.kr, usman@kut.ac.kr, jsg1215z@kut.ac.kr,eemohdusman@kut.ac.kr, jhryu@kut.ac.kr`

[2] Department of Mechanical Engineering, University of California, Santa Barbara, CA 93106 USA `ewhawkes@engineering.ucsb.edu`

[3] Department of Mechanical Engineering, Stanford University, Stanford, CA 94305 USA `aokamura@stanford.edu`

[4] on leave: Electrical Engineering Department, Faculty of Engineering (Shoubra), Benha University, Egypt `haitham.elhussieny@feng.bu.edu.eg`

978-1-5386-8094-0/18/$31.00 ©2018 IEEE

Because of its growing capability, controlling a vine-like robot in 3D space could be challenging using the input interfaces described above, especially from the first person view (FPV) perspective [19]. Due to the kinematic dissimilarity between these input interfaces and a vine robot (in shape, pose or degrees of freedom), the user's situational awareness could be negatively affected. One important problem is unintended "roll-over" of the robot head and its mounted camera due to the mis-perception of the robot's attitude [20]. When this happens, the human operator loses the sense of the robot orientation relative to its base or surroundings. Correcting this problem currently requires either highly experienced human operators or an automatic compensation mechanism that could increase system complexity.

One suggested solution for the roll-over problem is to use an input interface that has the same physical configuration (i.e. pose and shape) as the controlled robot [21]. However, designing an input interface that completely matches the kinematics of a growing vine robot is impractical due to length limitations. In this paper we propose a new flexible input interface with a fixed length that maps human bending commands into movements and provides an implicit shape sensing capability for a distal portion of the vine robot. The key features of the proposed design are (1) it has a similar flexible structure to the robot it controls, facilitating more intuitive commands; and (2) its compliance allows a human operator to understand the shape and magnitude of the bending commands sent to the robot, allowing a higher level of situational awareness and addressing the roll-over problem.

The rest of this paper is organized as follows. Section II presents the design of the new proposed flexible interface while Section III explains the mapping approaches proposed to map the input commands to robot movements. The experimental study is described in details in Section IV, while results are highlighted and discussed in section V. Finally, Section VI concludes the paper with remarks on the future implications.

II. The Proposed Flexible Interface

The goal of this research is to develop a flexible input interface that has a kinematic structure similar to the distal (foremost) part of the growing vine robot. As shown in Figure 2, an 18 cm flexible mesh structure is designed and 3D printed using the NinjaFlex [22] soft rubber material. This structure allows a soft passive return spring that serves as an implicit haptic feedback regarding the magnitude of the given command, with with potential to enhance situational awareness of the teleoperated robot and its environment [23].

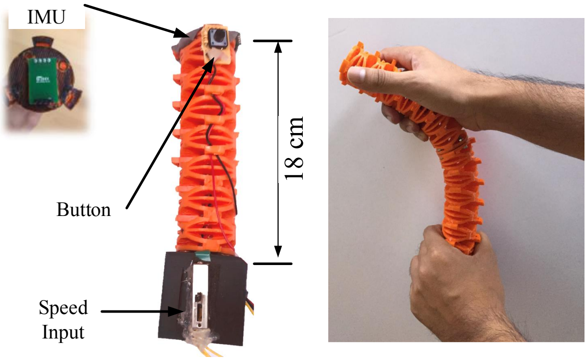


Fig. 2: Structure of the the proposed flexible input interface.

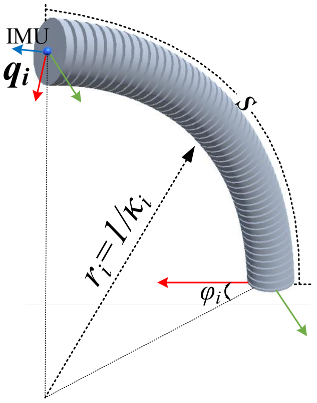


Fig. 3: Geometry of the proposed flexible interface with the input parameters, $\boldsymbol{q}_i$, κ_i, ϕ_i, and s used in the mapping techniques.

The input interface is equipped with an EBIMU-9DOFV3 IMU from E2BOX [24] which is mounted on the tip of the interface to measure the input orientation. In addition, a mechanical button is mounted for an additional mapping functionality. The growing speed of the vine robot can be controlled by a potentiometer mounted on the base of the interface. To measure the tip orientation relative to the base, an additional IMU could be mounted on the interface base. However, to reduce the degrees of freedom that the human has to control, the base is assumed to be fixed in the current design. The interface is normally handled parallel to the ground to mimic the pose of the growing vine robot.

Figure 3 shows the geometry of the proposed input interface. The only measured quantity is the tip orientation, represented as the unit quaternion $\boldsymbol{q}_i$. The curvature κ_i, and the angle of curvature, ϕ_i, are estimated from the tip orientation as well as the length of the interface, s.

III. Input Mapping Techniques

While the vine robot is growing with a certain speed v_r, steering movements are achieved by rotating the robot head in 3D space to reach a desired pose. Four mapping techniques are proposed to control the orientation of the robot head: position mode, position mode with indexing, rate mode, and shape mode.

A. Position Mode

In position mapping, the current measured tip orientation, $\boldsymbol{q}_i$, of the input interface is directly mapped to the robot head orientation, $\boldsymbol{q}_r$, with respect to global coordinates as follows

$$\boldsymbol{q}_r(t+1) = \boldsymbol{q}_i^{\alpha}(t), \qquad \alpha \in (0,1] \tag{1}$$

In the unit quaternion representation, the value $\alpha \in (0,1]$ determines the percentage of the input rotation that is sent to the robot head [25]. This value defines the input sensitivity, where the input orientation is fully mapped to the robot at $\alpha = 1$; at $0 < \alpha < 1$, a percentage of $\boldsymbol{q}_i$ will be executed.

B. Position Mode with Indexing

In this mode, the additional button is used to overcome the workspace limitation of the position mode. The current measured input orientation, $\boldsymbol{q}_i$, is mapped to the robot head rotation, $\boldsymbol{q}_r$, with respect a non-fixed reference frame, $\boldsymbol{q}_b$, as follows.

$$\boldsymbol{q}_r(t+1) = \boldsymbol{q}_b(t) * \boldsymbol{q}_i^{\alpha}(t), \qquad \alpha \in (0,1] \tag{2}$$

The reference frame $\boldsymbol{q}_b$ is incrementally updated to the recent robot head orientation while the button is not pressed, i.e. $k = 0$. It remains fixed to its previous value when the human operator presses and holds the button ($k = 1$) as follows.

$$\boldsymbol{q}_b(t) = k\boldsymbol{q}_b(t-1) + (1-k)\boldsymbol{q}_r(t) \tag{3}$$

This indexing allows the input to span a larger workspace by reinitializing the reference rotation $\boldsymbol{q}_b$. In addition, the decoupling between shaping the interface and executing the command allows humans to unbend the interface to its nominal shape after executing the command. This unintentional compensation happens upon release of the button; this prevents the communication of unwanted commands as occurs in the previous position mode.

C. Rate Mode

Rate mode control is a commonly used control strategy for teleoperation. In this mode, the orientation of the robot head with respect to its previous value is incrementally updated to the currently measured input as follows.

$$\boldsymbol{q}_r(t+1) = \boldsymbol{q}_r(t) * \boldsymbol{q}_i^{\alpha}(t), \qquad \alpha \in (0,1] \tag{4}$$

Here there is no direct physical mapping between the current orientation of the tip of the interface and the robot head orientation. Thus, if the input interface is held at a constant orientation, the robot will continue curving at constant curvature according to that input (or a fraction of the input, if $\alpha < 1$). This mode allows users to move the robot head through an infinite workspace without the additional input required by the position mode with indexing. Moreover, rate mode could act as a low-pass filter to remove sudden motions introduced by human operators. Because the robot updates its curvature rapidly, the sensitivity α has small values ($\alpha_{max} \approx 0.15$) in this mode compared to the two modes mentioned above.

D. Shape Mode

The previous three mapping strategies do not use the shape of the flexible input interface. In this mapping method, the shape of the input interface is used to control the vine robot movement.

The key components of the shape mapping mode are illustrated in Figure 4. First, the shape of the input interface in terms of the input curvature κ_i and the direction of curvature ϕ_i is estimated. Then, a curved path, $\mathbf{P}$, with the same curvature but at a different scale to the input, is generated as an array of 3D points that serves as a future path for the vine robot to follow. Since the robot is continuously growing in its forward direction, shape-following is achieved by simply changing the current heading angle of the vine robot toward the next point on the generated path.

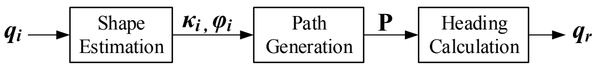
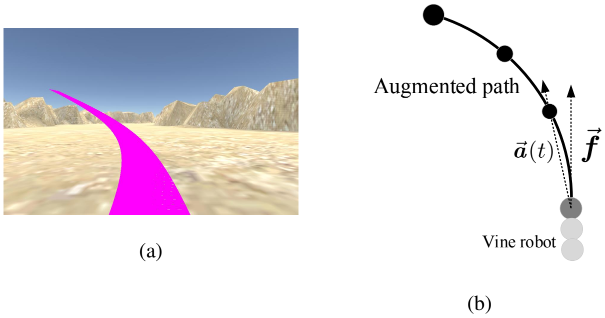


Fig. 4: Block diagram showing the key steps done in the shape mapping approach.



(a)

(b)

Fig. 5: (a) Augmentation of the generated future path **P** in advance to the vine robot head. (b) Heading rotation along the generated path.

Estimating the shape of the flexible input interface can be achieved by one of the shape sensing techniques surveyed in [26] for continuum robots. Here, based on the constant curvature assumption [1], the interface shape in terms of κ_i and ϕ_i can be retrieved by knowing its length, s, and measuring its tip rotation, $\boldsymbol{q}_i$. By this assumption, the input curve is represented as a segment of a circle circumference in 3D space which perhaps allows limited single curvature shapes. Compared to other continuum robots, multiple curvature shapes are not necessary in the case of vine robots. Since the robot will grow continuously following the commanded input, multiple curvatures can be achieved by commanding a sequence of single-curvature shapes.

Based on an inverse kinematic model of continuum robots [27], the shape parameters are calculated by equating the rotation matrix in terms of κ_i and ϕ_i, with the rotation matrix obtained from the measured input tip rotation $\boldsymbol{q}_i$, i.e. $\mathbf{R}(\kappa_i, \phi_i) = \mathbf{R}(\boldsymbol{q}_i)$. Thus, the curvature and its direction of the input interface are calculated as:

$$\kappa_i = \frac{cos^{-1}\left(1 - 2(q_{ix}^2 + q_{iy}^2)\right)}{s}, \qquad \kappa_i > 0 \tag{5}$$

$$\phi_i = tan^{-1}\left(\frac{q_{ix}q_{iw} + q_{iy}q_{iz}}{q_{ix}q_{iz} - q_{iy}q_{iw}}\right), \; -\pi \leq \phi_i \leq \pi \tag{6}$$

where q_{wi} and $[q_{xi}, q_{yi}, q_{zi}]^T$ are the scalar and the vector components of the measured input $\boldsymbol{q}_i$.

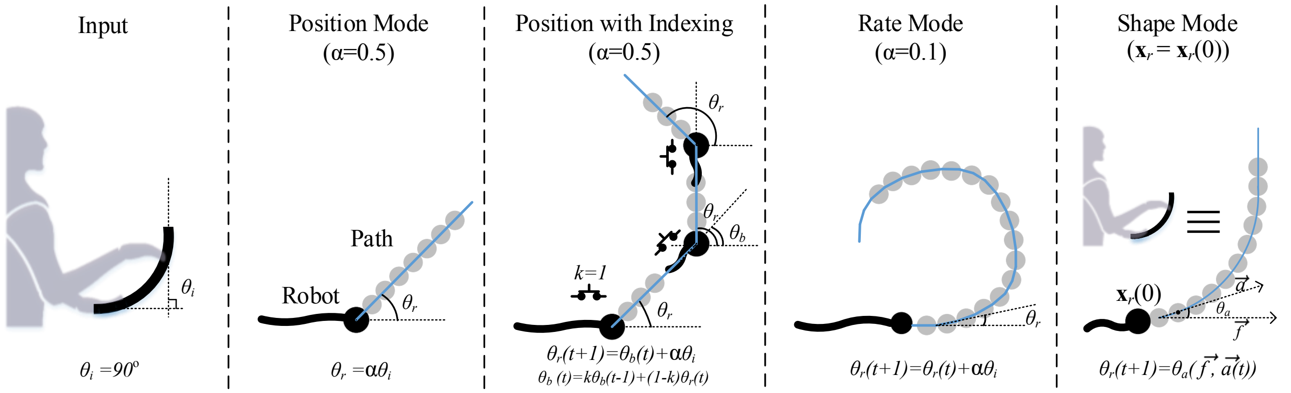

Input
Position Mode
(α=0.5)
Position with Indexing
(α=0.5)
Rate Mode
(α=0.1)
Shape Mode
($\mathbf{x}_r = \mathbf{x}_r(0)$)
θ_i
Path
Robot
θ_r
θ_r
θ_r
θ_b
k=1
θ_r
θ_r
$\mathbf{x}_r(0)$
$\vec{a}$
θ_a
$\vec{f}$
$\theta_i = 90^\circ$
$\theta_r = \alpha\theta_i$
$\theta_r(t+1)=\theta_b(t)+\alpha\theta_i$
$\theta_b(t)=k\theta_b(t-1)+(1-k)\theta_r(t)$
$\theta_r(t+1)=\theta_r(t)+\alpha\theta_i$
$\theta_r(t+1)=\theta_a(\vec{f},\vec{a}(t))$

Fig. 6: Comparison of the four proposed mapping approaches, where the input is the same in all cases (the interface tip is at a 90° angle with respect to the base) and the output is the path followed by the robot head.

Subsequently, the curved future path $\mathbf{P} = [\mathbf{Px}, \mathbf{Py}, \mathbf{Pz}]^T \in \mathbb{R}^{3\times M}$ is generated as a list of M Cartesian points in 3D from the retrieved shape parameters with a different length L. By solving the forward kinematics for a constant-curvature continuum model [27], this path is generated as follows.

$$\begin{bmatrix}\mathbf{Px}\\ \mathbf{Py}\\ \mathbf{Pz}\end{bmatrix} = \mathbf{x}_r(t-\delta) + \begin{bmatrix} cos\phi_i\left(\dfrac{cos(\kappa_i m)-1}{\kappa_i}\right)\\ sin\phi_i\left(\dfrac{cos(\kappa_i m)-1}{\kappa_i}\right)\\ \dfrac{sin(\kappa_i m)}{\kappa_i}\end{bmatrix} \quad (7)$$

where $\mathbf{x}_r(t-\delta) \in \mathbf{R}^3$ represents the robot head position at time $t-\delta$, where δ determines the degree of reactivity to any change in the input command, while m is calculated as

$$m = \left[\frac{L}{M}, 2\frac{L}{M}, 3\frac{L}{M}, \ldots, (M-1)\frac{L}{M}, L\right]$$

This generated path is augmented in advance to the vine robot head as a visual feedback about the future position of the robot as shown in Figure 5a. By considering the fact that the robot cannot penetrate the environment floor, any points calculated beneath are projected on the surface.

Finally, the points on the generated path are used to direct the vine robot heading angle by rotating its head toward the next point on the augmented path. Thus, the head rotation $\boldsymbol{q}_r$ is calculated as follows:

$$\boldsymbol{q}_r(t+1) = \boldsymbol{q}_a(\vec{f}, \vec{a}(t)) \quad (8)$$

where $\boldsymbol{q}_a(\vec{f}, \vec{a}(t))$ is the unit quaternion required to align the robot forward direction $\vec{\boldsymbol{f}}$ with the vector $\vec{\boldsymbol{a}}(t)$ directed toward the next point perceived at time t on the path as illustrated in Figure 5b.

In shape mapping, the user is not required to control the robot head in the other three orientation axes to reach a certain target as in the three mapping techniques mentioned above. Only he shape of the input interface is required, which could perhaps reduce the mental workload to compensate for the roll-over issue during vine robot teleoperation. In addition, with the help of the future path augmentation, the operator could easily plan the robot motion ahead of time which increases the maneuverability, especially in congested environments.

The key difference between shape mapping and other mapping technique is the way we interpret the interface input. In shape mode, we plan the future path the robot will follow toward reaching the pose of the interface tip. In contrast, the orientation of the robot head is immediately controlled by the pose of the interface tip in other mapping techniques according to the chosen frame of reference. A comparison between the four proposed mapping techniques is shown in Figure 6 in terms of the path followed by the robot head. As an example, a 90° up bending command around one axis is used to clearly compare the difference of each mapping method conceptually. Thus, all the previously mentioned unit quaternions are abstracted into a single angle notation θ.

IV. Experimental Study

A. Robot Simulator

To evaluate the proposed flexible input interface with its mapping techniques, a near-realistic vine robot is simulated to enable repeatability across conditions and users. In addition, a variety of unstructured environments could be incorporated with ease in simulation to evaluate the performance of the vine robot.

Recent advances in commercial off-the-shelf video-game physics engines have provided ever-improving realistic platforms which could be utilized to simulate a flexible, growing, vine-like robot in a realistic environment [28]. Unity 3D [29] was chosen to create a hyper-articulated robotic structure as shown in Figure 7. In addition to the visual effects, Unity facilitates the modeling of body collisions which in turn increases the realism of the simulation. The growing behavior of the simulated robot is realized by incrementally adding

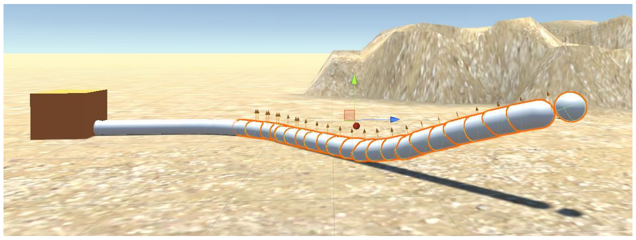


Fig. 7: Simulation of vine robot as a group of small connected rigid segments.

fixed-length, small segments behind the robot tip that are linked together through mass-spring dynamics.

B. Experiment Scenarios

For the purpose of the comparison study, two simulated environments were generated.

a) Valve reaching task: In this task, human subjects were asked to teleoperate the vine robot to reach a mechanical valve at the far end by navigating through a highly cluttered, unorganized industrial warehouse environment as shown in Figure 8. To guide subjects, a reference path leading to the valve was concurrently shown in the simulated environment. All subjects were asked to reach the valve as fast as they could by following the guidance path as accurately as they could.

b) Wrapping around task: In this task, the goal was to approach and climb a pillar by wrapping around it as shown in Figure 9, to mimic the behavior of vines found in nature. This task requires the operator to maintain a good sense of orientation to successfully wrap around the pillar toward its top. So, the aim of this scenario was to evaluate how the proposed interface could perform in the roll-over issue mentioned earlier. In order to discourage human operators from directly approach the pillar top, and instead make them wrap around it toward the top, a path was given around the pillar as a guidance.

C. Other Input Interfaces

A standard keyboard, a Logitech Extreme 3D Pro Joystick, and a Phantom Omni were used as other commercially available inputs to evaluate the performance of our proposed flexible interface. Proper mapping techniques were adopted for each interface as follows:

a) Keyboard: The left, right, up and down arrow keys of a standard keyboard were mapped to control the direction of the simulated vine robot into the four respective directions with a certain scale. Two additional buttons were used to either increase or decrease the growing speed of the robot.

b) Phantom Omni: The movement of the simulated robot was controlled using Cartesian coordinates of the pen stylus. The rate mapping mode was adopted, where the stylus translations in both horizontal and vertical directions result in turning the robot in horizontal and vertical directions accordingly. Additionally, the pushing direction of the stylus was chosen to change the growing speed of the robot. To provide a zero velocity reference, a returning force was generated using a virtual spring in x-, y- and z-axes using Open Haptics software toolkit [30]. Data communication between Unity3d and Open Haptics was done using ZMQ and its C# version netMQ libraries [31].

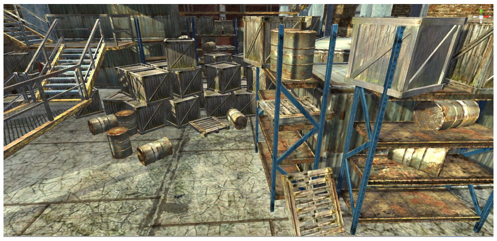


Fig. 8: First Person View (FPV) of the simulated cluttered environment.

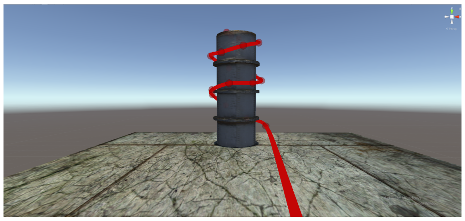


Fig. 9: The wrapping around experiment scenario used to evaluate the roll-over issue. The guidance path is shown in red.

c) Extreme 3D Pro Joystick: Both x and y directions of the joystick were used to simply rotate the vine robot in rate mode with a certain scale depending on the magnitude of inputs. The growing speed was controlled using the top up/down axes on the joystick.

D. Evaluation Metrics

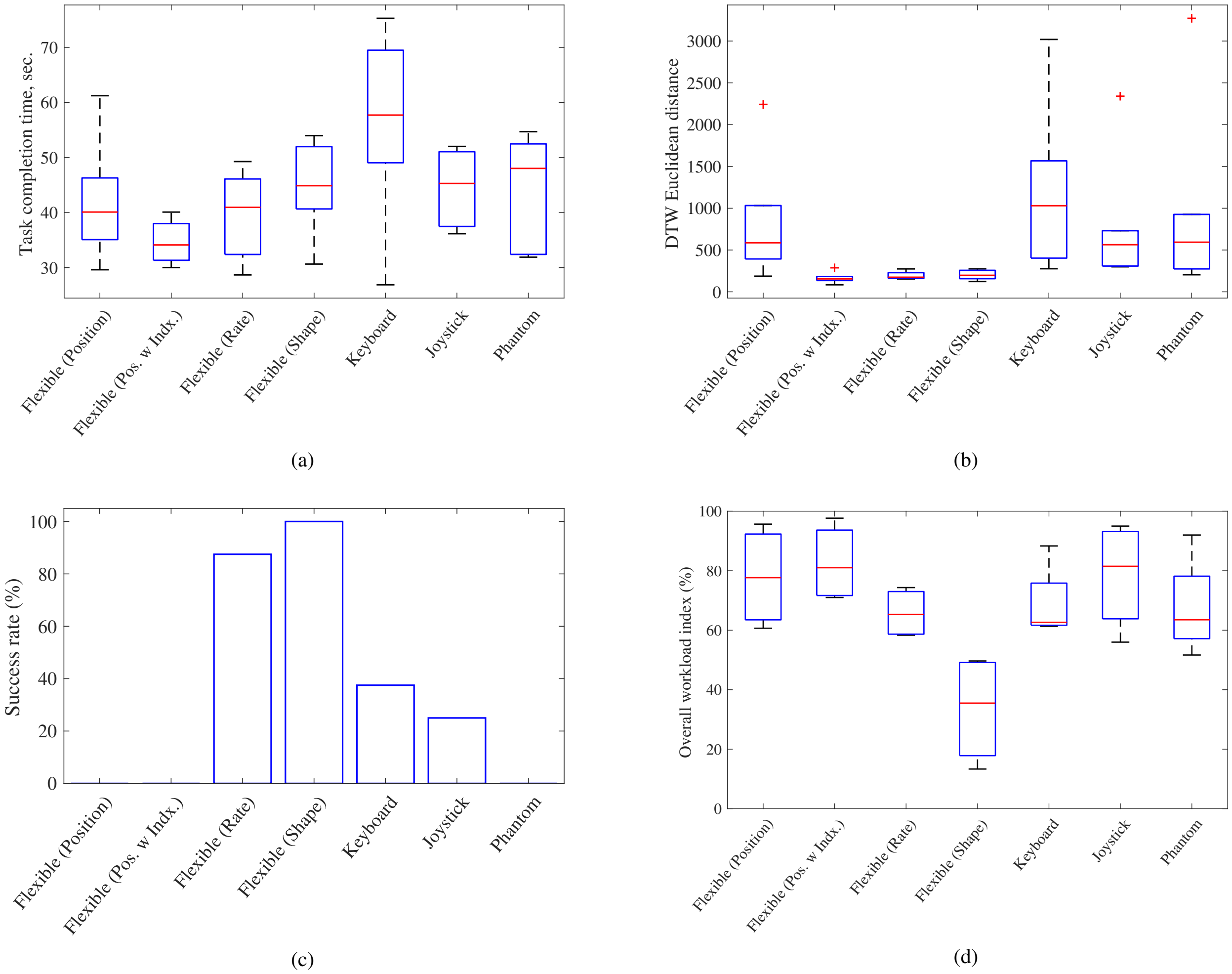
Three objective metrics and one subjective metric were used to assess the performance of each interface in accomplishing the simulated tasks.

a) Task Completion Time (TCT): The total time required to reach the target valve in the first scenario is measured in seconds. Accomplishing the task is identified once the human subject drives the robot head near the valve by a certain small distance.

b) Path accuracy: The deviation between the actual robot path and the guidance path is measured. In fact, the number of samples in a certain measured robot path is mainly affected by the time taken to complete the task. Thus, the well-known Dynamic Time Warping (DTW) [32] technique is used to find the optimal alignment between the two given (time-dependent) paths to properly measure the error between them. The sum of the Euclidean distances between corresponding points is measured after alignment.

c) Success rate: In the wrapping around scenario, the number of trials succeeded in reaching the top of the pillar is summed and normalized by the total number of trials for each input interface.

d) Subjective measure: As a subjective measure, the standard NASA Task Load Index (NASA-TLX) [33] was used. It is a widely used multidimensional, subjective assessment standard that rates perceived workload while achieving



(a)

(b)

(c)

(d)

Fig. 10: Performance comparison of the input interfaces: The newly developed flexible interface with its four mapping approaches, Keyboard, Joystick and Phantom Omni in terms of (a) Task completion time, (b) DTW Euclidean distance, (c) Success rate and (d) Overall workload index.

a particular task. For each input interface, subjects were asked four questions to rate the: mental load, temporal load, effort, and frustration scale of the wrapping around task. The overall workload index is calculated at the end as a one value metric based on the a weighted average scaling of the four mentioned indices.

Data was analyzed from six healthy adult males (25-33 years). All subjects were allowed three training runs on all the interfaces prior to data collection for both scenarios. The smoothness $\alpha = 0.8$ is chosen in both position and position with indexing mapping modes while $\alpha = 0.1$ is chosen in the rate mode of the flexible interface. Also, the reactivity factor $\delta = 0$ is chosen in shape mapping.

V. Results and Discussion

In the two experimental scenarios, the performance of the used input interfaces is highlighted in terms of the measured task completion time, DTW Euclidean distance, success rate and the overall task workload in Figure 10. For each interface, the median, interquartile range and max/min values are displayed.

As observed, in terms of task completion time (Figure 10a), the proposed flexible interface, especially with its indexed position mapping, outperformed other interfaces in general in the valve reaching scenario. This could account for the increase in the degree of subjects' situation awareness, allowing them to safely drive the robot to its maximum speed toward the target. In fact, this is most probably due to the mentioned implicit shape sensing capability that the flexible interface has, compared to the other input interfaces. Moreover, in terms of the path following accuracy (Figure 10b), the proposed flexible interface shows better performance in position with indexing, rate and shape modes compared to others due to the ability of continuously commanding the vine robot movements. However, due to its limited workspace, the flexible interface with position mode has shown comparable results to that of joystick and Phantom Omni.

In the wrapping-around scenario, the success rate of each

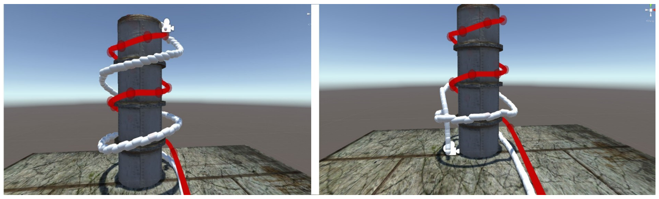


Fig. 11: (Left) A successful wrapping-around task achieved by the flexible interface in shape mapping, while the robot (in white) has failed to reach the pillar top by warping along the guidance path (red) using the Phantom Omni input (Right) due to the lack of orientation awareness (roll-over).

input interface is highlighted in Figure 10c. As noted, higher success rates of 100% and 90% are achieved by the flexible interface in both rate and shape modes, respectively. In rate mode, this, is accounted for the ability the human operator has to compensate for the roll-over issue that could happen on the robot head while wrapping around toward top of the pillar. This compensation is facilitated by controlling the orientation of the robot head in roll, pitch, and yaw angle. However, in shape mapping, human subjects were not required to do this kind of roll-over compensation since the robot is continuously adjusting its heading angle based on the given future path as discussed in Section III. Examples of both a successful and a failed wrapping-around trials are shown in Figure 11.

As expected, the robot did not successfully achieved the designated wrapping task with the flexible interface in position mapping. Due to the workspace limitation of this mode, the robot was not able to complete a circular path to turn around the pillar. However, it was surprising that even with indexing mode, the simulated robot did not achieve this task. The reason could account for the shape of the chosen pillar. Its small radius might impose a difficulty to rapidly press the button, bend the interface and then release the button continuously to make the robot wrap successfully around it. For the same reason, we could account for the small success rates of other keyboard and joystick inputs, where the roll-over could be insignificant while maneuvering the robot in short radii turns toward reaching a small height pillar.

The NASA-TLX subjective assessment in terms of the overall workload is shown in Figure 10d for each of the input interfaces. As a consequence of all the satisfactory results in the previously mentioned metrics, the flexible interface with shape mapping has ranked subjectively to have a low index of workload. This expected result could perhaps be accounted due to two factors. The first factor is the way we compensate for the roll-over issue in this mode. Human subjects were allowed to intuitively wrap around the pillar with no need to have a sense of the current robot orientation. Second, subjects were easily compensating the robot movements not only based on the magnitude of bending they already made to the interface but also to the future path the robot will take. The unintentional fusion of both the current and the future state of the robot helps in achieving the given task. It is particularly interesting that the wrapping around task was achieved with ease in shape mode by simply bending the interface similar to the foremost part of the robot.

It could be questionable that if we introduce this kind of future path augmentation in the first three mapping techniques, would they perform better or not. We believe that the future path augmentation in the other three modes will not provide too much useful information since in these modes we are controlling the immediate pose of the robot head. Thus, a tilted line will be always shown on the front camera which will go straight immediately once the robot execute the given command. On the other hand, providing a curved path with the same shape mode philosophy to other three modes will not even help. As mentioned in Section III, the orientation of the interface tip is differently interpreted in these proposed modes. Thus, there is no guarantee if the robot will follow this curved path.

VI. Conclusion

In this paper, a flexible input interface is proposed to intuitively teleoperate the new type of soft growing, vine-like robot. The developed interface was designed to have a soft passive returning as well as sharing similar kinematics with the growing robot, as an attempt to increase the human situational awareness in general and to solve the roll-over issue in particular.

Four command mapping strategies are proposed to map the interface bending commands into robot movements; namely, *position*, *position with indexing*, *rate* and *shape* mapping. The performance of the newly developed flexible interface is evaluated compared with that of other commercially available interfaces. Two simulation based experiments have conducted with a near-realistic simulated vine robot to assess the capability of our proposed interface in tackling the mentioned lack of orientation issue that could happen while moving the robot in 3D space.

The proposed flexible interface, especially with the shape mapping technique, has shown significant improvements toward solving the mentioned roll-over issue compared to that of either other mapping techniques or other input interfaces. This roll-over free capability has proven through conducting one of the key motions required in vine-robot; reaching by wrapping around. As noted, a 100% success rate was achieved by using the shape mapping technique in the proposed flexible input interface. In addition, a subjective NASA-TLX assessment has conducted and the proposed interface in its shape mapping has achieved a reduced order of overall task overload.

The results we have obtained encourage us to go through more than one future direction to enhance the proposed interface. Definitely, it could be evaluated in more challenging vine-robot scenarios, such as opening a valve, tightening a knot or even manipulation of small objects. Another direction is to investigate how the performance of the proposed flexible interface could be affected if more than one flexible section is appended to it in achieving complex motion commands.

ACKNOWLEDGMENTS

The authors thank Laura H. Blumenschein and Joseph D. Greer for their contributions to vine robot design, modeling, and control.

REFERENCES

[1] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[2] M. Yim, K. Roufas, D. Duff, Y. Zhang, C. Eldershaw, and S. Homans, "Modular reconfigurable robots in space applications," *Autonomous Robots*, vol. 14, no. 2, pp. 225–237, 2003.

[3] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "A review on modeling, implementation, and control of snake robots," *Robotics and Autonomous Systems*, vol. 60, no. 1, pp. 29–40, 2012.

[4] E. W. Hawkes, L. H. Blumenschein, J. D. Greer, and A. M. Okamura, "A soft robot that navigates its environment through growth," *Science Robotics*, vol. 2, no. 8, p. eaan3028, jul 2017. [Online]. Available: http://robotics.sciencemag.org/lookup/doi/10.1126/scirobotics.aan3028

[5] M. B. Wooten and I. D. Walker, "A novel vine-like robot for in-orbit inspection." 45th International Conference on Environmental Systems, 2015.

[6] A. Sadeghi, A. Mondini, E. Del Dottore, V. Mattoli, L. Beccai, S. Taccola, C. Lucarotti, M. Totaro, and B. Mazzolai, "A plant-inspired robot with soft differential bending capabilities," *Bioinspiration and Biomimetics*, vol. 12, no. 1, p. 015001, 2017.

[7] D. Mishima, T. Aoki, and S. Hirose, "Development of pneumatically controlled expandable arm for search in the environment with tight access," in *Field and Service Robotics*. Springer, 2003, pp. 509–518.

[8] H. Tsukagoshi, N. Arai, I. Kiryu, and A. Kitagawa, "Smooth creeping actuator by tip growth movement aiming for search and rescue operation," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1720–1725.

[9] A. Mikawa, H. Tsukagoshi, and A. Kitagawa, "Tube actuator with drawing out drive aimed for the inspection in the narrow and curved path," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2010, pp. 1368–1373.

[10] L. H. Blumenschein, A. M. Okamura, and E. W. Hawkes, "Modeling of bioinspired apical extension in a soft robot," in *Conference on Biomimetic and Biohybrid Systems*. Springer, 2017, pp. 522–531.

[11] G. Niemeyer, C. Preusche, and G. Hirzinger, "Telerobotics," in *Springer Handbook of Robotics*. Springer, 2008, pp. 741–757.

[12] C. E. Lathan and M. Tracey, "The effects of operator spatial perception and sensory feedback on human-robot teleoperation performance," *Presence: Teleoperators & Virtual Environments*, vol. 11, no. 4, pp. 368–377, 2002.

[13] C. Fellmann, D. Kashi, and J. Burgner-Kahrs, "Evaluation of input devices for teleoperation of concentric tube continuum robots for surgical tasks," in *Proc. SPIE*, vol. 9415, 2015, pp. 94 151O–1.

[14] T. K. Morimoto, J. J. Cerrolaza, M. H. Hsieh, K. Cleary, A. M. Okamura, and M. G. Linguraru, "Design of patient-specific concentric tube robots using path planning from 3-d ultrasound," in *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2017, pp. 165–168.

[15] A. Majewicz and A. Okamura, "Cartesian and joint space teleoperation for nonholonomic steerable needles," in *IEEE World Haptics Conference*, 2013, pp. 395–400.

[16] C. G. Frazelle, A. D. Kapadia, K. E. Fry, and I. D. Walker, "Teleoperation mappings from rigid link robots to their extensible continuum counterparts," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2016, pp. 4093–4100.

[17] R. M. Scott, "Continuum surrogate software interface for teleoperation of continuum robots," Ph.D. dissertation, Clemson University, 2016.

[18] M. Csencsits, B. A. Jones, W. McMahan, V. Iyengar, and I. D. Walker, "User interfaces for continuum robot arms," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems,(IROS 2005)*. IEEE, 2005, pp. 3123–3130.

[19] K. M. Stanney, R. R. Mourant, and R. S. Kennedy, "Human factors issues in virtual environments: A review of the literature," *Presence*, vol. 7, no. 4, pp. 327–351, 1998.

[20] J. Y. Chen, E. C. Haas, and M. J. Barnes, "Human performance issues and user interface design for teleoperated robots," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1231–1245, 2007.

[21] D. E. McGovern, "Experience and results in teleoperation of land vehicles," in *Pictorial Communication in Virtual and Real Environments*. Taylor & Francis, Inc., 1991, pp. 182–195.

[22] "NinjaFlex 3D," https://ninjatek.com/, [Online; accessed 25-February-2018].

[23] N. A. A. Martin, V. Mittelstdt, M. Prieur, R. Stark, and T. Br, "Passive haptic feedback for manual assembly simulation," *Procedia CIRP*, vol. 7, pp. 509 – 514, 2013.

[24] "E2box," www.e2box.co.kr, [Online; accessed 25-February-2018].

[25] E. B. Dam, M. Koch, and M. Lillholm, *Quaternions, interpolation and animation*. Datalogisk Institut, Københavns Universitet Copenhagen, 1998, vol. 2.

[26] C. Shi, X. Luo, P. Qi, T. Li, S. Song, Z. Najdovski, T. Fukuda, and H. Ren, "Shape sensing techniques for continuum robots in minimally invasive surgery: A survey," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 8, pp. 1665–1678, 2017.

[27] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.

[28] J. Rieffel, D. Knox, S. Smith, and B. Trimmer, "Growing and evolving soft robots," *Artificial Life*, vol. 20, no. 1, pp. 143–162, 2014, pMID: 23373976.

[29] "2018 Unity Technologies," https://unity3d.com/, [Online; accessed 25-February-2018].

[30] "3dsystems openhaptics," https://www.3dsystems.com/haptics-devices/openhaptics, [Online; accessed 25-February-2018].

[31] "Netmq," https://netmq.readthedocs.io, [Online; accessed 25-February-2018].

[32] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.

[33] S. G. Hart and L. E. Staveland, "Development of NASA-TLX (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.