2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Model-Based Estimation of the Gravity-Loaded Shape and Scene Depth for a Slim 3-Actuator Continuum Robot with Monocular Visual Feedback

Yuyang Chen, Shu'an Zhang, Lingyun Zeng,
Xiangyang Zhu, *Member, IEEE* and Kai Xu*, *Member, IEEE*

***Abstract*—Fruitful developments on continuum robots have been witnessed in recent years due to their movements and manipulation capabilities in confined spaces. Due to the nature that a continuum robot has an infinite number of DoFs (Degrees of Freedom), majority of the existing systems deployed abundant actuators such that the robot can be controlled in separately modeled and actuated segments with constant or variable curvature. As the shape of a continuum robot is always jointly determined by its actuation and the interactions from the environment, it is hence worth exploring the opposite approach that how a task can be accomplished with a minimal number of actuators. This paper presents the first step of such an investigation where a slim 3-actuator continuum robot is actuated to reach different spatial locations under gravity. As the gravity greatly affects the robot's shape, a monocular camera, together with two UKFs (Unscented Kalman Filters), was used to concurrently estimate the robot's shape and the feature depth. Then the estimated shape can be used in updating the kinematics model of the robot to achieve motion control. Experiments were conducted to validate the effectiveness of the proposed shape estimation, which promises the motion control implementation in the near future work.**

I. Introduction

DUE to the structural compactness, motion compliance and dexterity, continuum robots have been widely applied in medical and industrial applications [1]. Because of the nature that a continuum robot has an infinite number of DoFs (Degrees of Freedom), majority of the existing systems deployed abundant actuators such that the robot can be controlled in separately modeled and actuated segments. The modeling approaches for these segments include i) rigid links approximation [2, 3], ii) assumed constant curvature [4-6], and iii) variable curvature formulation [7-9].

Manuscript received September 15, 2018. This work was supported in part by the National Natural Science Foundation of China (Grant No. 51435010, Grant No. 51722507 and Grant No. 91648103), and in part by National Key R&D Program of China (Grant No. 2017YFC0110800).

Yuyang Chen and Shu'an Zhang are with the RII Lab (Lab of Robotics Innovation and Intervention), UM-SJTU Joint Institute, Shanghai Jiao Tong University, Shanghai, 200240, China (emails: supandoria@sjtu.edu.cn and jimmyonthego@sjtu.edu.cn).

Lingyun Zeng, Xiangyang Zhu and Kai Xu are with the State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China (asterisk indicates the corresponding author, emails: me_maxqi@sjtu.edu.cn, mexyzhu@sjtu.edu.cn and k.xu@sjtu.edu.cn).

Unless an infinite number of actuators with unlimited capabilities are used, the shape of a continuum robot is always jointly determined by its actuation and the interactions from the environment. For example, the slim continuum robot loaded by its gravity has a shape that greatly deviates from the assumed shape under the constant curvature assumption, as shown in Fig. 1. However, the robot's actual (complex) shape in Fig. 1 does not entirely prevent the slim continuum robot from performing a task (e.g., reach different spatial locations). It is hence worth exploring how such a task can be accomplished with a minimal number of actuators.

This paper presents the first step of an investigation on how a slim 3-actuator (two for bending and one for translation) continuum robot can be controlled to reach different spatial locations under gravity. As the shape is greatly deflected by gravity, a shape sensing modality has to be provided, unless the gravitational direction, accurate structural parameters and sophisticated nonlinear mechanics formulation are provided.

Shape sensing modalities mainly utilizes electromagnetic (EM) tracking, fiber Bragg gratings (FBG) sensing and image processing [10]. For example, using EM tracking, a recursive least square algorithm was used to estimate the shape of a concentric-tube robot [11], while an EKF (Extended Kalman Filter) was used to estimate the shape of a multi-segment continuum manipulator [12].

While EM sensors are sensitive to electronic noises and FBG sensors need complicated implementation and relatively expensive apparatus for demodulation, an on-board monocular camera is chosen, as externally placed cameras are not realistically practical for the use in confined spaces.

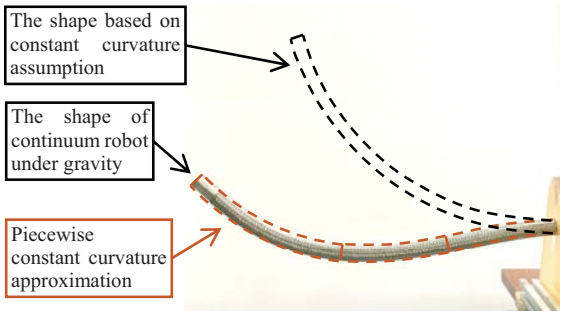

Fig. 1. Substantial shape deviations of a slim 3-actuator continuum robot under gravity can be observed; the actual shape may be iteratively estimated using a piecewise constant curvature model.

The views from the on-board monocular camera under the robot's actuation indirectly indicate the shape variations of the robot. Hence, two UKFs (Unscented Kalman Filters) were used to concurrently estimate the robot's shape and the feature depth. The UKFs are chosen over EKFs (Extended Kalman Filters) because the UKF may provide better estimation accuracy without using the system's Jacobian matrix [13, 14].

Although it is also possible to achieve the control of a continuum manipulator via model-free estimation approaches that primarily concern the robots' Jacobian [15, 16], a model-based approach, which utilizes the understanding of the system under control, may produce more information to increase the self-awareness of the environment and the system, as well as achieve the motion control. For example, the deflected shape of the slim continuum manipulator in Fig. 1 can be approximated by three "virtual segments" with constant curvature. Then two UKFs were used to concurrently estimate the robot's shape and the feature depth. The estimated shape can be sequentially used in updating the kinematics model of the robot to achieve motion control.

This paper is organized as follows. Kinematics of the slim continuum manipulator was derived in Section II. Then the estimation problem using a dual-UKF formulation is elaborated in Section III. In Section IV, design of the continuum robot is described and the experimental results are presented. The conclusion and the future works are summarized in Section V.

II. Kinematics Model

The 3-actuator continuum robot under investigation consists of a tip camera, a slim continuum body and an actuation unit, as in Fig. 2 and Fig. 4.

The continuum body consists of i) one central tube, ii) four backbones, iii) one end ring, iv) one base ring, and v) many spacer rings. The rings are all attached to the central tube equidistantly, while the four backbones are fixed to the end ring and can slide inside the holes in the spacer rings and the base ring. The backbones are driven in a differential push-pull mode to realize 2-DoF bending. The robot possesses a translational DoF as it can be linearly fed by a linear actuator in the actuation unit.

The robot's shape deflected by gravity can be approximated as three stacked virtual segments, as shown in Fig. 1 and Fig. 2. Modeling of the continuum body and the tip camera is presented in Section II.A and Section II.B, respectively.

A. Kinematics of the Virtual Segments

A schematic of the 3-actuator continuum robot is in Fig. 2. The robot is modeled as three stacked virtual segments with constant curvature to provide an approximation close enough to the robot's actual shape under gravity. Each segment has 2 DoFs for bending and 1 DoF for length variation. As the robot moves, these shape variables of each virtual segment would be estimated by the UKFs as in Section III.

The kinematics model for each segment can be derived with the constant curvature assumption that is widely used [4, 5].

The nomenclature is summarized in Table I and the coordinates are plotted in Fig. 2 and defined as follows.

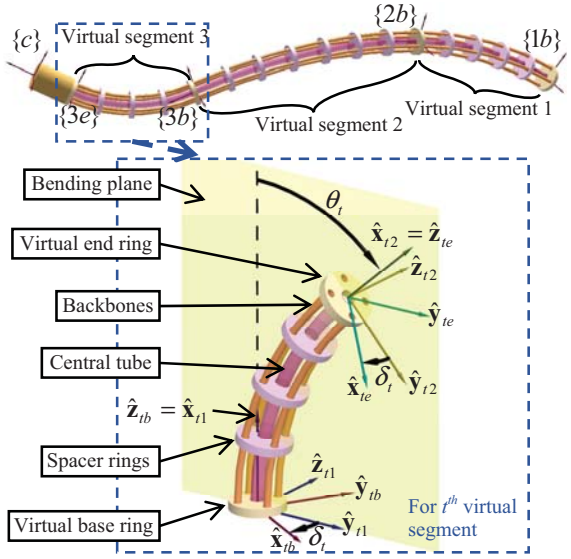

Fig. 2. Nomenclature and coordinate definitions for the continuum robot.

TABLE I
NOMENCLATURE USED IN THIS KINEMATICS MODEL

Symbol	Definition
t	Index of the virtual segments, $t = 1, 2, 3$.
i	Index of the backbones, $i = 1, 2, 3, 4$. Numbering of the segments always precedes that of the backbones.
k	Index of the discrete-time sequence, $k = 1, 2, 3, \cdots$. For a concise notation, k is marked as superscripts.
φ	Division angle of the backbones along the pitch circle. $\varphi = \pi/2$.
r	Distance between a backbone and the ring center.
θ_t	Bending angle of the t^{th} virtual segment. θ_t is defined as the angle rotating from $\hat{\mathbf{x}}_{t1}$ to $\hat{\mathbf{x}}_{t2}$ in the bending plane.
δ_{ti}	For the t^{th} virtual segment, a right-handed rotation angle from $\hat{\mathbf{y}}_{t1}$ about $\hat{\mathbf{x}}_{t1}$ to a ray passing through the i^{th} backbone.
δ_t	$\delta_t \equiv \delta_{t1}$ and $\delta_{ti} = \delta_t + (i-1)\varphi$.
L_t, L_{ti}	Lengths of the central tube and the i^{th} backbone in the t^{th} virtual segment, respectively.
L_S, L_{Si}	Total length of the central tube and the i^{th} backbone of the robot (spanning all virtual segments), respectively.
q_{ti}	$q_{ti} \equiv L_{ti} - L_t$ is the actuation length for the i^{th} backbones in t^{th} virtual segment.
$\boldsymbol{\psi}_t$	$\boldsymbol{\psi}_t = [\theta_t\ \delta_t\ L_t]^T$ is the configuration vector defining the shape of the t^{th} virtual segment.
${}^{tb}\mathbf{R}_{te}$	Coordinate transformation matrix from frame $\{te\}$ to frame $\{tb\}$. $\{te\}$ coincides with $\{(t+1)b\}$.
${}^{tb}\mathbf{p}_{te}$	Position vector pointing from the t^{th} virtual base ring center to the center of the t^{th} virtual end ring (namely the virtual base ring in the $(t+1)^{th}$ virtual segment).

- *Base Ring Coordinate*, $\{tb\} = \{\hat{\mathbf{x}}_{tb}, \hat{\mathbf{y}}_{tb}, \hat{\mathbf{z}}_{tb}\}$, is attached to the virtual base ring in the t^{th} virtual segment. $\hat{\mathbf{x}}_{tb}$ points from the center of the ring to the first backbone and $\hat{\mathbf{z}}_{tb}$ is normal to the ring. $\{1b\}$ is designated as the world frame.
- *Bending Plane Coordinate 1*, $\{t1\} = \{\hat{\mathbf{x}}_{t1}, \hat{\mathbf{y}}_{t1}, \hat{\mathbf{z}}_{t1}\}$, is

attached to the virtual base ring. The virtual segment bends in its XY plane.

- *Bending Plane Coordinate 2*, $\{t2\}=\{\hat{\mathbf{x}}_{t2},\hat{\mathbf{y}}_{t2},\hat{\mathbf{z}}_{t2}\}$, is obtained by rotating $\{t1\}$ about $\hat{\mathbf{z}}_{t1}$, translating its origin to the center of the virtual end ring. $\hat{\mathbf{x}}_{t2}$ is normal to the ring.
- *End Ring Coordinate*, $\{te\}=\{\hat{\mathbf{x}}_{te},\hat{\mathbf{y}}_{te},\hat{\mathbf{z}}_{te}\}$, is attached to the virtual end ring. The $\hat{\mathbf{z}}_{te}$ coincides with $\hat{\mathbf{x}}_{t2}$, while $\hat{\mathbf{x}}_{te}$ points from the center to the first backbone.
- *Camera Coordinate*, $\{c\}=\{\hat{\mathbf{x}}_c,\hat{\mathbf{y}}_c,\hat{\mathbf{z}}_c\}$, is attached to the tip camera. $\{c\}$ is translated from $\{3e\}$.

For each virtual segment, the configuration can be written in terms of the actuation lengths as follows.

$$\begin{cases}\delta_t=\operatorname{atan2}(q_{t2},-q_{t1})\\ \theta_t=(q_{t2}\sin\delta_t-q_{t1}\cos\delta_t)/r\end{cases} \tag{1}$$

The forward kinematics can be derived from the constant curvature assumption, as in Eq. (2) and Eq. (3).

$${}^{tb}\mathbf{p}_{te}=\frac{L_t}{\theta_t}\begin{bmatrix}(1-\cos\theta_t)\cos\delta_t & (\cos\theta_t-1)\sin\delta_t & \sin\theta_t\end{bmatrix}^T \tag{2}$$

Where ${}^{tb}\mathbf{p}_{te}=[0\ 0\ L_t]^T$ when $\theta_t=0$.

$${}^{tb}\mathbf{R}_{te}=\operatorname{Rot}(\hat{\mathbf{z}}_{tb},-\delta_t)\operatorname{Rot}(\hat{\mathbf{z}}_{t1},\theta_t)\operatorname{Rot}(\hat{\mathbf{z}}_{te},\delta_t) \tag{3}$$

Where $\operatorname{Rot}(\hat{\mathbf{n}},\gamma)$ is a rotation matrix about $\hat{\mathbf{n}}$ by an angle γ.

The direct kinematics for the three-segment robot can be written from the single-segment kinematics:

$${}^{1b}\mathbf{R}_{3e}={}^{1b}\mathbf{R}_{1e}\,{}^{2b}\mathbf{R}_{2e}\,{}^{3b}\mathbf{R}_{3e} \tag{4}$$

$${}^{1b}\mathbf{p}_{3e}={}^{1b}\mathbf{p}_{1e}+{}^{1b}\mathbf{R}_{1e}\,{}^{2b}\mathbf{p}_{2e}+{}^{1b}\mathbf{R}_{1e}\,{}^{2b}\mathbf{R}_{2e}\,{}^{3b}\mathbf{p}_{3e} \tag{5}$$

The three virtual segments constitute the entire continuum robot. Hence the following constraints on the total backbones and central tube lengths hold.

$$L_S=L_1+L_2+L_3,\ L_{Si}=L_{1i}+L_{2i}+L_{3i}\ (i=1,2) \tag{6}$$

B. Frontal Projection Model of the Tip Pinhole Camera

In order to simplify the derivation, a *frontal projection model* of a pinhole camera is used as in Fig. 3. In this model, the camera aperture is simplified as a point (a.k.a. the pinhole) where the camera frame $\{c\}$ locates. The image plane is put in front of the pinhole to avoid image flipping [14].

The optical axis of the camera is orthogonal to the image plane and $\hat{\mathbf{z}}_c$ coincides with the optical axis. $\hat{\mathbf{x}}_c$ and $\hat{\mathbf{y}}_c$ are parallel to the pixel coordinates $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$. The mapping from a 3D point in the world frame ${}^{1b}\mathbf{p}_g=[X\ Y\ Z]^T$, to the 2D point $\mathbf{z}_g=[u\ v]^T$ in the pixel coordinate, is given as follows.

$$\begin{bmatrix}\mathbf{z}_g\\1\end{bmatrix}=\frac{1}{d_c}\mathbf{M}\mathbf{T}\begin{bmatrix}{}^{1b}\mathbf{p}_g\\1\end{bmatrix} \tag{7}$$

Where d_c is the unknown depth of the target point in the camera coordinate $\{c\}$.

$\mathbf{T}$ is referred to as the extrinsic parameters of the camera. It is the homogeneous transformation matrix that relates the world coordinate $\{1b\}$ to the camera coordinate $\{c\}$.

$$\mathbf{T}=\begin{bmatrix}{}^{1b}\mathbf{R}_c^T & -{}^{1b}\mathbf{R}_c^T\,{}^{1b}\mathbf{p}_c\end{bmatrix} \tag{8}$$

Where ${}^{1b}\mathbf{R}_c$ and ${}^{1b}\mathbf{p}_c$ are the rotation matrix and position vector of $\{c\}$ w.r.t. $\{1e\}$. Given the configuration vectors of each virtual segment, $\mathbf{T}$ can be obtained from Eqs. (1) to (5).

$\mathbf{M}$ is referred to as the intrinsic parameters of the camera and is formulated as follows.

$$\mathbf{M}=\begin{bmatrix}f_u & 0 & u_0\\0 & f_v & v_0\\0 & 0 & 1\end{bmatrix} \tag{9}$$

Where f_u and f_v are the focal lengths expressed in pixel dimensions in the $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ axes, respectively. u_0 and v_0 are the offsets of the image origin from the intersection between the optical axis and the image plane. The intrinsic parameters can be obtained through a camera calibration.

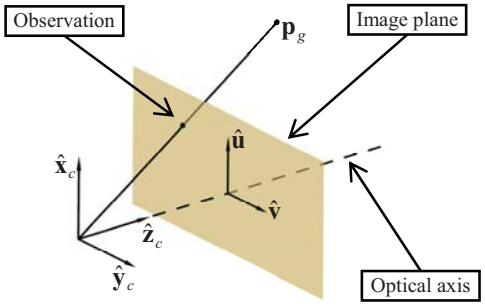

Fig. 3. The frontal projection model of the tip pinhole camera

III. Shape and Depth Estimation

The unscented Kalman filter (UKF) is employed to solve the estimation problem in this paper. Referring to the kinematics model, the shape of the robot can be determined by 9 variables (three for each virtual segment: θ_t, δ_t and L_t). A set of nine variables will be designated as the state variables to estimate the status of the robot as in Section III.A. Then, the UKF estimation is presented in detail in Section III.B.

A. System State-Space Representation

For a typical discrete-time nonlinear system, the state vector $\mathbf{x}^k$, the input $\boldsymbol{\mu}^k$ and the output $\mathbf{y}^k$ are related as follows.

$$\mathbf{x}^k=g(\mathbf{x}^{k-1},\boldsymbol{\mu}^k)+\mathbf{r}^k \tag{10}$$

$$\mathbf{y}^k=h(\mathbf{x}^k,\mathbf{d})+\mathbf{q}^k \tag{11}$$

Where g and h represent the system dynamic model. $\mathbf{r}^k$ and $\mathbf{q}^k$ are the zero-mean Gaussian process noise and observation noise whose covariances are given by $\mathbf{R}^k$ and $\mathbf{Q}^k$.

For the continuum robot discussed in this paper, the following set of nine variables is used as the state vector $\mathbf{x}^k$.

$$\mathbf{x}^k=[q_{11}^k\quad q_{12}^k\quad \rho_1^k\quad \cdots\quad q_{31}^k\quad q_{32}^k\quad \rho_3^k]^T \tag{12}$$

Where q_{t1} and q_{t2} correspond to θ_t and δ_t using Eq. (1); ρ_t^k is defined from the ratio of the t^{th} virtual segment length to the total length, mapping the ratios to the required value ranges.

$$\rho_t^k=\tan\left(3\pi\left(L_t^k/L_S^k-1/3\right)\right) \tag{13}$$

The input vector $\boldsymbol{\mu}^k$ is defined to include the incremental actuation lengths for the backbones and the central tube.

$$\boldsymbol{\mu}^k=\begin{bmatrix}L_{S1}^k-L_{S1}^{k-1} & L_{S2}^k-L_{S2}^{k-1} & L_S^k-L_S^{k-1}\end{bmatrix}^T \tag{14}$$

Given the current input $\boldsymbol{\mu}^k$ and state $\mathbf{x}^k$, the next state $\mathbf{x}^{k+1}$ is determined by the *prediction model g* as formulated below.

$$\begin{cases} \rho_t^{k+1} = \rho_t^k \\ q_{ti}^{k+1} = \dfrac{q_{ti}^k + L_t^k}{L_S^k} L_S^{k+1} - L_t^{k+1} \end{cases} \quad (t = 1,2,3; \quad i = 1,2) \tag{15}$$

Equation (15) is constructed based on the postulation that the input $\boldsymbol{\mu}^k$ does not change the ratio of backbone (central tube) lengths in each virtual segment to the total backbone (central tube) lengths. The assumption is essential since it provides a prediction model for the state-transition.

In Eq. (11), $\mathbf{y}^k$ is the vector of all sensor outputs. It includes all the pixel coordinates of *m* feature points as well as the encoder outputs from the actuators that drive the total actuation lengths of the backbones and the central tube.

$$\begin{aligned} \mathbf{y}^k &= \begin{bmatrix} \left(\mathbf{z}_1^k\right)^T & \cdots & \left(\mathbf{z}_m^k\right)^T & L_{S1}^k & L_{S2}^k & L_S^k \end{bmatrix}^T \\ &= \begin{bmatrix} u_1^k & v_1^k & \cdots & u_m^k & v_m^k & L_{S1}^k & L_{S2}^k & L_S^k \end{bmatrix}^T \end{aligned} \tag{16}$$

Therefore, the output model *h*, is the mapping from the state vector $\mathbf{x}^k$ to the image coordinates $\mathbf{z}^k$ and the total actuation lengths L_S and L_{Si}. The former can be described by the kinematics model in Eq. (1) to Eq. (5) and the camera model in Eq. (7) to Eq. (9), while the latter represents the length constraints in Eq. (6) that the backbones and the central tube in each virtual segment must comply. Because the depth information is missing in the camera model, $h(\cdot, \mathbf{d})$ is a nonlinear function with unknown parameters $\mathbf{d}$.

B. The UKF-Based Estimation

The state estimation problem for the nonlinear system described in Eq. (10) and Eq. (11) is solved using the UKF for the reasons stated in Section I.

As any Kalman filter based estimator, an iteration in the UKF state estimation consists of two steps, i.e., the time prediction and the measurement update. Before the time prediction, the *unscented transform* is performed to obtain the σ-points $\boldsymbol{\chi}^{k-1}$, given the state expectation $\mathbf{x}^{k-1}$ and covariance matrix $\mathbf{P}^{k-1}$

$$\boldsymbol{\chi}^{k-1} = \begin{bmatrix} \mathbf{x}^{k-1} & \mathbf{x}^{k-1} \pm \left(\sqrt{(n+\lambda)\mathbf{P}^{k-1}}\right)^{[j]} \end{bmatrix} \tag{17}$$

Where *n* is the dimension of vector $\mathbf{x}^{k-1}$. $j = 1, 2, \cdots, n$. $[j]$ in the superscript denotes the j^{th} column of the matrix. Then $\boldsymbol{\chi}^{k-1}$ is an n by $(2n+1)$ matrix. Each column is a σ-point representing a specific sample from the Gaussian distribution. $\lambda = \alpha^2(n + \kappa) - n$ is a scaling factor. α and κ are constants determining the distribution of the sampled σ-point.

Next, two sets of weights, $\boldsymbol{\omega}_m$ and $\boldsymbol{\omega}_c$ are determined for further calculations of means and covariances respectively:

$$\begin{cases} \boldsymbol{\omega}_m^{[0]} = \dfrac{\lambda}{\lambda + n}, \quad \boldsymbol{\omega}_c^{[0]} = \dfrac{\lambda}{\lambda + n} + (1 - \alpha^2 + \beta) \\ \boldsymbol{\omega}_m^{[j]} = \boldsymbol{\omega}_c^{[j]} = \dfrac{\lambda}{2(\lambda + n)} \end{cases} \tag{18}$$

Where β is a constant that represents the prior knowledge of the distribution of $\mathbf{x}$.

In the time prediction step, the σ-points are directly transmitted through the nonlinear functions *g* and *h* to obtain the expected outputs $\tilde{\mathbf{y}}^k$.

$$\overline{\boldsymbol{\chi}}^{k*} = g\left(\boldsymbol{\chi}^{k-1}, \boldsymbol{\mu}^k\right) \tag{19}$$

$$\overline{\mathbf{x}}^k = \sum_{j=0}^{2n} \boldsymbol{\omega}_m^{[j]} \overline{\boldsymbol{\chi}}^{k*[j]} \tag{20}$$

$$\overline{\mathbf{P}}^k = \sum_{j=0}^{2n} \boldsymbol{\omega}_c^{[j]} (\overline{\boldsymbol{\chi}}^{k*[j]} - \overline{\mathbf{x}}^k)(\overline{\boldsymbol{\chi}}^{k*[j]} - \overline{\mathbf{x}}^k)^T + \mathbf{R}^k \tag{21}$$

$$\overline{\boldsymbol{\chi}}^k = \begin{bmatrix} \overline{\mathbf{x}}^{k-1} & \overline{\mathbf{x}}^{k-1} \pm \left(\sqrt{(n+\lambda)\overline{\mathbf{P}}^{k-1}}\right)^{[j]} \end{bmatrix} \tag{22}$$

$$\overline{\mathbf{Y}}^k = h\left(\overline{\boldsymbol{\chi}}^k, \mathbf{d}\right) \tag{23}$$

$$\tilde{\mathbf{y}}^k = \sum_{j=0}^{2n} \boldsymbol{\omega}_m^{[j]} \overline{\mathbf{Y}}^{k[j]} \tag{24}$$

In the measurement update step, the Kalman gain $\mathbf{K}^k$ is calculated based on the covariance of observation $\mathbf{P}_{\mathbf{y},\mathbf{y}}^k$ and the cross-covariance of state-observation $\mathbf{P}_{\mathbf{x},\mathbf{y}}^k$:

$$\mathbf{P}_{\mathbf{y},\mathbf{y}}^k = \sum_{j=0}^{2n} \boldsymbol{\omega}_c^{[j]} (\overline{\mathbf{Y}}^{k[j]} - \tilde{\mathbf{y}}^k)(\overline{\mathbf{Y}}^{k[j]} - \tilde{\mathbf{y}}^k)^T + \mathbf{S}^k \tag{25}$$

$$\mathbf{P}_{\mathbf{x},\mathbf{y}}^k = \sum_{j=0}^{2n} \boldsymbol{\omega}_c^{[j]} (\overline{\boldsymbol{\chi}}^{k[j]} - \overline{\mathbf{x}}^k)(\overline{\mathbf{Y}}^{k[j]} - \tilde{\mathbf{y}}^k)^T \tag{26}$$

$$\mathbf{K}^k = \mathbf{P}_{\mathbf{x},\mathbf{y}}^k \left(\mathbf{P}_{\mathbf{y},\mathbf{y}}^k\right)^{-1} \tag{27}$$

$$\mathbf{x}^k = \overline{\mathbf{x}}^k + \mathbf{K}^k \left(\mathbf{y}^k - \tilde{\mathbf{y}}^k\right) \tag{28}$$

$$\mathbf{P}^k = \overline{\mathbf{P}}^k - \mathbf{K}^k \mathbf{P}_{\mathbf{y},\mathbf{y}}^k \left(\mathbf{K}^k\right)^T \tag{29}$$

Equations (17) to (29) specify an entire UKF iteration, which recursively gives the estimation of the state variable $\mathbf{x}^k$.

Please note that in Eq. (23), the depth $\mathbf{d}$ is unknown. The use of the tip camera causes the incorporated unknown feature depth. In this scenario, both the state variables and the feature depth shall be estimated, which involves the approach of *dual estimation* [13].

In the dual UKF algorithm, two UKFs are running concurrently, and the system states and feature depths are updated sequentially. The first UKF is used for the state estimation of the robot described by Eq. (10) and Eq. (11). The second UKF estimates the feature depth $\mathbf{d}$ by assuming it as an evolving variable.

$$\mathbf{d}^k = \mathbf{d}^{k-1} + \mathbf{r}_d^k \tag{30}$$

$$\mathbf{z}^k = h\left(\tilde{\mathbf{x}}^k, \mathbf{d}^k\right) + \mathbf{q}_d^k \tag{31}$$

Where $\mathbf{r}_d^k$ and $\mathbf{q}_d^k$ are the Gaussian white noises with covariances $\mathbf{R}_d^k$ and $\mathbf{Q}_d^k$. Eq. (31) differs from Eq. (11) in that $\mathbf{d}^k$ is now regarded as a variable of the function *h*. Thus, a similar process following Eq. (17) to Eq. (29) gives the depth estimation $\tilde{\mathbf{d}}^k$, which in return, is used in the first UKF.

IV. Experimental Validation

To examine the performance of the UKF estimation presented in Section III, a series of experiments were performed on the prototype as shown in Fig. 4.

The robot is first described in Section IV.A, while the experimental verifications are reported in Section IV.B.

A. System Description

The robot was used to verify the proposed estimation, as in Fig. 4. A camera was installed on the distal tip of the robot, an actuation unit drives the continuum robot and a computer is used for motion control as well as the online estimation.

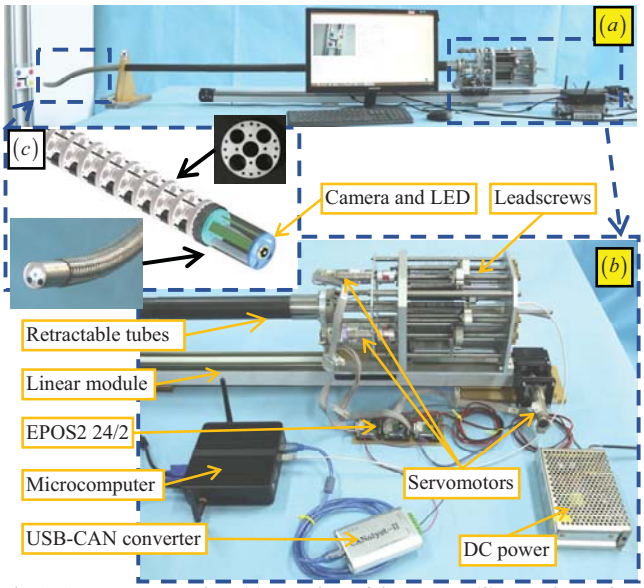

Fig. 4. System construction: (*a*) overview of the system, (*b*) actuation and control infrastructure, and (*c*) design of eye-in-hand monocular continuum manipulator.

The continuum robot has a total length of 2 meters and an outer diameter of 23 mm. A stainless-steel-woven outer tube wraps over the continuum body to form a smooth surface.

The spacer rings are 3 mm thick with an outer diameter of 21 mm. Several holes are drilled for passing the backbones and the central tube. The holes for the backbones are arranged on the same pitch circle with a radius r = 9.25 mm and the division angle $\varphi = \pi/2$.

The central tube has an outer diameter of 6 mm, which matches the diameter of the central hole in the spacer rings. Both the spacer rings and the central tube are made from polyethylene (PE). They are fixed together by ultrasonic plastic welding so as to improve the torsional rigidity.

A miniaturized camera (Guangmiao Electronics, China) with 6 LEDs is integrated on the distal tip. It is 7 mm in diameter and 27 mm in length. The camera is connected to the computer through a USB cable. It provides 30-fps video stream with a resolution of 640×480.

The backbones are made from super-elastic nitinol rod with a 1-mm diameter. The backbones are driven by four leadscrews. A linear module with an effective travel of 1.5 meters translates the entire robot. Six carbon fiber retractable tubes are used to keep a portion of the manipulator straight.

Three servomotors (DCX22L, Maxon Inc.) with the GPX-22 gearheads (gear ratio of 21:1) and the ENX16 encoders (512 Counts per Turn), are used to drive the leadscrews and the linear module. The servomotors are controlled by three Maxon EPOS2 24/2 digital controllers. An industrial microcomputer with a Celeron J1900 4-core CPU serves as the central controller of the system. It communicates with the EPOS2 by CAN buses through the USB-CAN converter (CANalyst-II, Chuangxin Tech. Inc., China).

B. Experimental Verifications

In the verification experiments, the robot was linearly fed out to reach a few spatial locations, simulating an inspection motion. The robot was greatly deflected by gravity.

The continuum robot starts to move from a short and straight pose with a 60 mm extruded length. The initial pose is assumed to be known. Four colored dots were used as fixed feature points in space. The intrinsic parameters of the camera were obtained by a calibration following the method in [17]. The pixel coordinates of the colored points are obtained through a filter in HSV space using OpenCV 2.7.

To evaluate the estimation performance, an optical tracker (Micron Tracker SX60, Claron Technology Inc.) was used to obtain the actual positions of the feature points, as well as the actual shape of the continuum robot, as demonstrated in Fig. 5(a). The estimation algorithm ran on the microcomputer. The shape estimation results were exported and plotted in MATLAB (The MathWorks Inc).

The experimental verification process is presented in Fig. 5(*b-e*). The robot first performed an insertion and steering motion, then it periodically swung around a position.

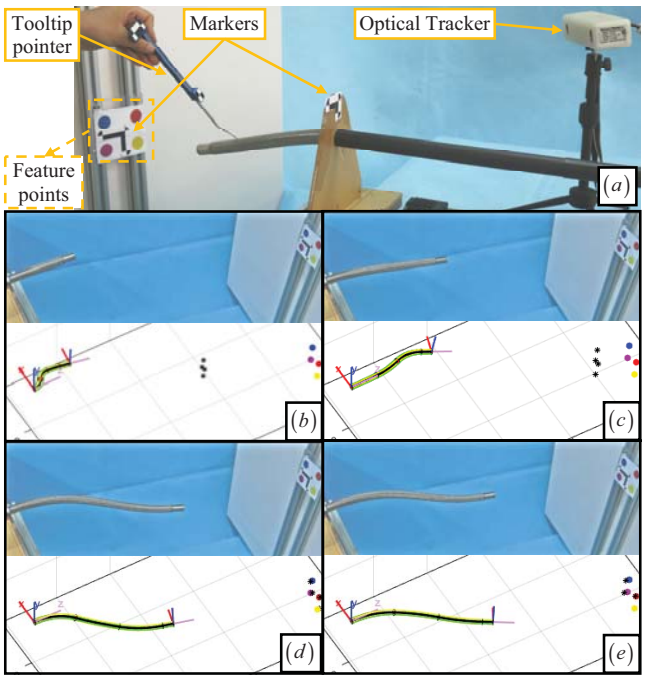

Fig. 5. (*a*) Experiment setup, and (*b*) – (*e*) experimental verification processes and the corresponding shape and depth estimations.

The shape estimation deviates from the actual robot shape in the beginning, as in Fig. 5. (*b*) and (*c*). But when the depth estimation errors diminished as the robot swung to obtain more data, the estimated and actual shapes become close. The estimations with different initial guesses of the feature depth

were examined to be comparable. The process is shown in the multimedia extension.

The estimation results after 2500 iterations are plotted in Fig. 6. Each iteration took 30ms, limited by the frame rate of the tip camera. The shape estimation has a 10.6-mm maximal error along the extruded 550 mm of the robot. On the other hand, a convergence of the estimated depths to the actual values can be seen from Fig. 7. The maximal steady-state depth estimation error is 9.4 mm.

Overall, the adopted dual UKFs generated the estimation results with acceptable accuracy. A brief discussion on the errors is as follows. The adopted piecewise constant curvature model is an approximation of the actual shape. It inherently introduces errors to the estimation. Furthermore, an ideal camera model was assumed for the feature depth estimation. A poor estimation of the feature depth can result in large shape estimation errors.

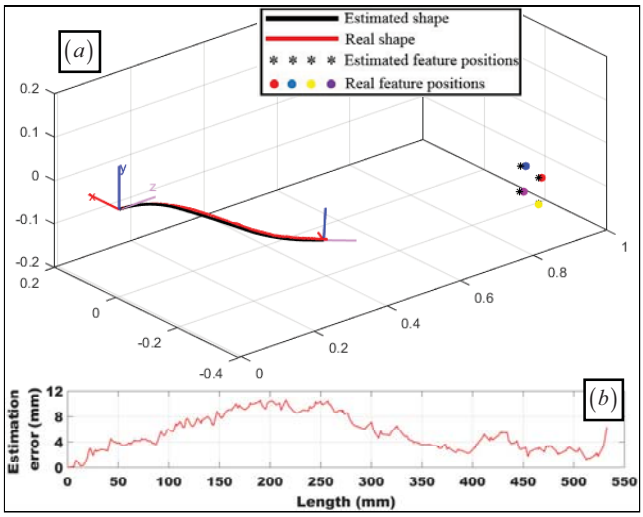

Fig. 6. Estimation results: (*a*) the actual shape and depth estimation are plotted together; (*b*) the shape estimation errors along the robot's length.

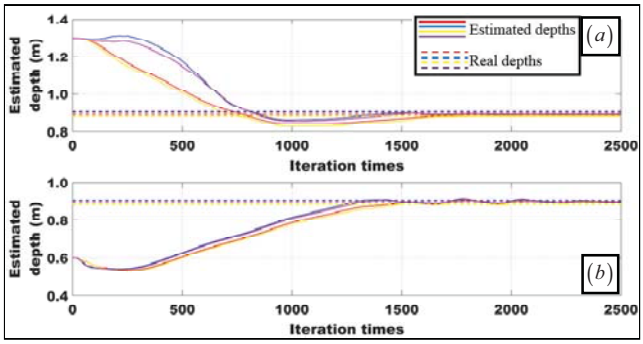

Fig. 7. Depth estimation results converge from different initial depth values in (a) and (b).

V. Conclusions and Future Works

This paper presents a model-based estimation for the scene depths and the gravity-affected shape of a slim 3-actuator continuum robot, using a dual UKFs approach.

Based on the piecewise constant curvature kinematics model, the robot's shape and the feature depths are estimated concurrently. Acceptable results were obtained in the experimental verifications: maximal estimation errors for the shape deviation and the feature depths of 10.6 mm and 9.4 mm, respectively.

With the estimated shape, the kinematics of the robot can be updated such that motion control of this continuum robot in constrained spaces can be carried out in a near future.

References

[1] I. D. Walker, "Continuous Backbone "Continuum" Robot Manipulators," *ISRN Robotics,* vol. 2013, No.726506, pp. 1-19, 2013.

[2] G. S. Chirikjian and J. W. Burdick, "A Modal Approach to Hyper-Redundant Manipulator Kinematics," *IEEE Transactions on Robotics and Automation,* vol. 10, No.3, pp. 343-354, June 1994.

[3] T. Zheng, D. T. Branson, E. Guglielmino, R. Kang, G. A. M. Cerda, M. Cianchetti, M. Follador, I. S. Godage, and D. G. Caldwell, "Model Validation of an Octopus Inspired Continuum Robotic Arm for Use in Underwater Environments," *ASME Journal of Mechanisms and Robotics,* vol. 5, No.2, p. 021004, Mar 2013.

[4] R. J. Webster and B. A. Jones, "Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review " *International Journal of Robotics Research,* vol. 29, No.13, pp. 1661-1683, Nov 2010.

[5] K. Xu and N. Simaan, "Analytic Formulation for the Kinematics, Statics and Shape Restoration of Multibackbone Continuum Robots via Elliptic Integrals," *Journal of Mechanisms and Robotics,* vol. 2, No.011006, pp. 1-13, Feb 2010.

[6] P. E. Dupont, J. Lock, B. Itkowitz, and E. Butler, "Design and Control of Concentric-Tube Robots," *IEEE Transactions on Robotics,* vol. 26, No.2, pp. 209-225, April 2010.

[7] D. C. Rucker and R. J. Webster, "Statics and Dynamics of Continuum Robots With General Tendon Routing and External Loading," *IEEE Transactions on Robotics,* vol. 27, No.6, pp. 1033-1044, Dec 2011.

[8] W. S. Rone and P. Ben-Tzvi, "Continuum Robot Dynamics Utilizing the Principle of Virtual Power," *IEEE Transactions on Robotics,* vol. 30, No.1, pp. 275-287, Feb 2014.

[9] T. Mahl, A. Hildebrandt, and O. Sawodny, "A Variable Curvature Continuum Kinematics for Kinematic Control of the Bionic Handling Assistant," *IEEE Transactions on Robotics,* vol. 30, No.4, pp. 935-949, Aug 2014.

[10] 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, Aug 2017.

[11] C. Kim, S. C. Ryu, and P. E. Dupont, "Real-time Adaptive Kinematic Model Estimation of Concentric Tube Robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Hamburg, Germany, 2015, pp. 3214-3219.

[12] A. Ataka, P. Qi, A. Shiva, A. Shafti, H. Wurdemann, H. Liu, and K. Althoefer, "Real-Time Pose Estimation and Obstacle Avoidance for Multi-Segment Continuum Manipulator in Dynamic Environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, 2016, pp. 2827-2832.

[13] E. A. Wan and R. v. d. Merwe, "The Unscented Kalman Filter," in *Kalman Filtering and Neural Networks*, S. Haykin, Ed., 2001, pp. 221-282.

[14] T. D. Barfoot, *State Estimation for Robotics*. New York, NY, USA: Cambridge University Press, 2017.

[15] M. C. Yip and D. B. Camarillo, "Model-Less Feedback Control of Continuum Manipulators in Constrained Environments," *IEEE Transactions on Robotics,* vol. 30, No.4, pp. 880-889, 2014.

[16] M. Li, R. Kang, D. T. Branson III, and J. S. Dai, "Model-free Control for Continuum Robots Based on an Adaptive Kalman Filter," *IEEE/ASME Transactions on Mechatronics,* vol. 23, No.1, pp. 286-297, Feb 2018.

[17] Z. Zhang, "Flexible Camera Calibration by Viewing a Plane from Unknown Orientations," in *IEEE International Conference on Computer Vision (ICCV)*, Kerkyra, 1999, pp. 666-673.