2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Vision Based Flexible Beam Tip Point Control

Yunjun Xu and Erich Ritz

Abstract—**Because of the light weight and less wear and tear on components, the flexible beam has been and will continue to be an appealing option for civil and military applications. However, flexibility brings with it unwanted oscillations and severe chattering which may even lead to an unstable system. To tackle these challenges, a two-time scale controller is presented to track a desired tip point signal and at the same time mitigate the tip point vibration. To obtain more precise information of the tip point location and facilitate the easy extension to multiple-flexible-link problems, a camera is used to provide vision feedback in which the delayed vision signal is compensated by the state estimator and predictor. The controller is experimentally verified, and shown to exceed the performance of other tested controllers.**

Index Terms—**Flexible beam, vision based control, vibration control, robust control, sliding mode control.**

I. Introduction

THE control of flexible beams has been a heavily-studied problem, with applications to space structures [1], automated manufacturing, assembly tasks [2], flexible XY positioning systems in aerial ladder trucks [3], and many others that require a lightweight but long-reach arm [4]. By allowing the use of flexible structures, lighter parts can be used, leading to lower energy consumption, less wear and tear, safer operations [5], and lower cost [6]. However, new challenges are introduced in that motion is no longer restricted to the joints which connect members, and oscillations in the structure may increase the settling time in commanding the structure to the desired position. Hence it is inevitable that the control bandwidth required for precision pointing will excite the flexible structure modes.

In controlling the flexible beam, different sensors, such as strain gauges [3], cameras [7-9], optical sensors [10], and accelerometers [11] have all been used.

One common method of measuring the beam tip position is the combination of a motor sensor to keep track of the root position of the beam and one or more strain gauges on the beam itself to measure deflection from its rigid body state [5] [12]. Based on this kind of measurement, different control methods have been investigated. For example, Kojima [1] uses an adaptive deflection-limiting input shaping technique, where the control signal is conditioned to minimize the excitation of the flexible modes of the structure. The maximum bending moment in the beam is cut in half, and the residual vibration can roughly be reduced by 75% for desired set points instead of tracking a time-varying desired signal. To handle the unmodeled dynamics and uncertainties associated with the system, the sliding mode control has been used by Etxebarria [5], Kaynak [13], Korondi [14], and Madhavan [15]. In the approach by [5], an LQR scheme is augmented for the fast mode and a boundary layer augmented sliding mode control is applied for the slow mode. By adding the LQR control for the fast flexible mode, oscillations in the beam are damped in 1 second and better tracking of the slow mode is achieved. However in Etxebarria's approach, a larger overshoot occurs which may come from the unbounded control gain.

Because the tip point information is not directly measured from the strain gauge, a model is needed to relate measurements to the tip deflection. Additionally, due to wave propagation along the length of the beam, the tip point response occurs slightly after a control input [7]. Strain gauges located near the root will give a near-immediate response to control action, which when extrapolated to the tip point position, gives an incorrect result. To achieve a more accurate measurement, sensor averaging and actuator averaging method has been proposed by Weng, Lu, and Trumper [6]. However, multiple sensors and actuators have to be used along the flexible beam and the advantages of the light weight will be compromised.

A more direct way of measuring the tip point is to use a vision based system. A camera mounted at the root of the beam measures the deflection from the centerline of the image [9]. The main drawback here is the mass of the camera is added to the structure that the motor must move, reducing the advantages of using flexible structures. The camera may also be placed in a stationary position to track the absolute position of a target. Stieber [8] used a single stationary camera to view the two dedicated visual targets; one was the payload target, and the other, the berth target. In this method, the only added mass to the system is the visual target. The relative position between the two was computed to provide necessary information for guidance. As shown in [8], the target converges to a new set point of 1 radian in about 2 seconds.

Instead of using cameras, optical sensors can also be used [10] for the purpose of simplicity although they are very susceptible to noise. The controller and sensing method presented here aim to accomplish several goals. First, after an initial calibration the camera can spot a point target and relay its position without the need for reference objects. Unlike [16], the delay in the camera signal is corrected

Y. Xu is with the Department of Mechanical, Materials, and Aerospace Engineering, University of Central Florida, Orlando, FL 32816 USA. Phone: 407-823-1745; Fax: 407-823-0208; (e-mail: yjxu@ou.edu).

E. Ritz is with Zona Technology Inc., Scottsdale, AZ 82258 USA; (email: erich@zonatech.com)

explicitly by a state estimator and predictor, which also eliminates overshoot caused by delayed measurements. Because the dynamics in the control method presented are separated into slow and fast modes, the slow control can be based on pre-existing controllers for rigid manipulators. Based upon an enhanced sliding mode controller, the control method not only tracks step inputs, but also time-varying inputs as well.

The remainder of this paper is organized as follows. Section II describes how the two-timescale robust control is designed based on the separated flexible dynamics. The vision based tip point estimation is described in Section III, which includes the camera calibration, target point prediction, and Kalman filter based signal delay compensation. In Section IV, the experiment setup is listed and results are shown. We conclude the paper in Section V.

II. Multi-Timescale Robust Control Design

A multi-timescale robust controller is presented to track a commanded tip-point signal. The dynamics are split into slow and fast subsystems; the slow system tracks the desired signal and the fast system minimizes vibrations in the beam.

A. Flexible Beam Dynamics

The plant is described by the differential equation

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}u + \boldsymbol{\xi} \tag{1}$$

and the measurement is modeled as

$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{\eta} \tag{2}$$

where the states are $\boldsymbol{x} = [\theta, \alpha, \dot{\theta}, \dot{\alpha}]^T \in \Re^{4\times 1}$ (motor angle θ, beam deflection α, and their corresponding rates as shown in Fig. 1). The input u is the voltage command to the motor. $\boldsymbol{\xi}$ and $\boldsymbol{\eta}$ are independent white noise, with covariance matrices of $\boldsymbol{Q}$ and $\boldsymbol{R}$ respectively. $\boldsymbol{A} \in \Re^{4\times 4}$, $\boldsymbol{B} \in \Re^{4\times 1}$, and $\boldsymbol{C} \in \Re^{2\times 1}$ are constants state, input and output matrices. Note that the given model accounts for only one flexible mode of the beam. In actuality, the beam's shape is a superposition of an infinite number of mode shapes; however, it was found in the experiments that only the first mode contributed significantly to the tested beam tip deflection.

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -K_{rr} & -K_{rf} & -V_{rr} & -V_{rf} \\ -K_{fr} & -K_{ff} & -V_{fr} & -V_{ff} \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ B_r \\ B_f \end{bmatrix} \tag{3}$$

$$\boldsymbol{Q} = \begin{bmatrix} 3.386\times10^{-6} & 1.0\times10^{-10} & 1.0\times10^{-10} & 1.0\times10^{-10} \\ 1.0\times10^{-10} & 3.386\times10^{-8} & 1.0\times10^{-10} & 1.0\times10^{-10} \\ 1.0\times10^{-10} & 1.0\times10^{-10} & 0.0036 & 1.0\times10^{-10} \\ 1.0\times10^{-10} & 1.0\times10^{-10} & 1.0\times10^{-10} & 0.0004 \end{bmatrix} \tag{4}$$

$$\boldsymbol{R} = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \tag{5}$$

where K is the mass normalized stiffness and V is the mass normalized Coriolis and centrifugal coefficient. The subscripts specify how that term relates the rigid (r) and flexible (f) modes, respectively. For example, the K_{rf} term is the coefficient on the flexible mode, α, in the equation for the rigid mode $\ddot{\theta}$. The coefficients for the selected flexible beam model are shown in the following $\boldsymbol{A}$ and $\boldsymbol{B}$ matrices as

$$\boldsymbol{A} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 438.2 & -50.33 & 0 \\ 0 & -841.4 & 55.42 & -1.628 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ 77.72 \\ -86.13 \end{bmatrix} \tag{6}$$

There are two measurements used in this work: case (1) y_1: motor encoder that measures $\dot{\theta}$ and $\boldsymbol{C}_1 = [0 \;\; 0 \;\; 1 \;\; 0]$; case (2) y_2: a camera that measures γ and $\boldsymbol{C}_2 = [1 \;\; 1 \;\; 0 \;\; 0]$. Therefore, the input matrix is defined as

$$\boldsymbol{C} = \left[\boldsymbol{C}_1^T, \boldsymbol{C}_2^T\right]^T \tag{7}$$

The covariance matrices associated with the measurements noise $\boldsymbol{\eta} = [\eta_1, \eta_2]^T$ are $R_1 = 3.386\times10^{-6}$ and $R_2 = 1.000\times10^{-10}$.

Since the camera measurement is delayed by τ (roughly 120 ms), the measurement y_2 is rewritten as

$$y_2(t) = \boldsymbol{C}_2\boldsymbol{x}(t-\tau) + \eta_2(t-\tau) \tag{8}$$

B. State Estimator

Two Kalman filters are used to obtain the optimal state estimates. Here the method of estimating $\hat{\boldsymbol{x}}_1$ through the motor measurement is shown, whereas the method of finding $\hat{\boldsymbol{x}}_2$ using an augmented predictor for the delay compensation is described in Section III.D. Note that the subscripts "1" and "2" in the states represent the cases that the states are estimated based on measurements "1" and "2", respectively.

For the non-delayed measurement (based on the motor), the estimated state $\hat{\boldsymbol{x}}_1$ is found from the Kalman filter as

$$\dot{\hat{\boldsymbol{x}}}_1 = (\boldsymbol{A} - \boldsymbol{K}_1\boldsymbol{C}_1)\hat{\boldsymbol{x}}_1 + \boldsymbol{B}u + \boldsymbol{K}_1 y_1 \tag{9}$$

where $\boldsymbol{K}_1 = \boldsymbol{P}_1\boldsymbol{C}_1^{\mathrm{T}}\boldsymbol{R}_1^{-1}$ and $\boldsymbol{P}_1$ is the solution of the Riccati equation described as

$$\dot{\boldsymbol{P}}_1 = \boldsymbol{A}\boldsymbol{P}_1 + \boldsymbol{P}_1\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{Q} - \boldsymbol{P}_1\boldsymbol{C}_1^{\mathrm{T}}R_1^{-1}\boldsymbol{C}_1\boldsymbol{P}_1 \tag{10}$$

With an initial condition of

$$\boldsymbol{P}_1(0) = \begin{bmatrix} 1\times10^{-3} & 6.83\times10^{-13} & 6.11\times10^{-12} & -9.53\times10^{-11} \\ 6.83\times10^{-13} & 3.41\times10^{-7} & 2.54\times10^{-6} & 9.35\times10^{-7} \\ 6.11\times10^{-12} & 2.54\times10^{-6} & 5.04\times10^{-5} & 1.60\times10^{-5} \\ -9.53\times10^{-11} & 9.35\times10^{-7} & 1.60\times10^{-5} & 1.60\times10^{-4} \end{bmatrix} \tag{11}$$

Note here $\boldsymbol{P}_1(0) = \boldsymbol{0}$ works well but with a longer settling time if compared with the one provided in Eq. (11).

C. Timescale Separation

As one of the methods in controlling under-actuated

systems, the dynamics equation is decomposed into slow and fast modes based upon the singular perturbation theory [17]. A small scale factor is defined as $\varepsilon^2 = k_m^{-1}$, where k_m^{-1} is the smallest stiffness constant (for the selected testbed here, $k_m = 0.0025$), and a new fast state is introduced as

$$\psi = \alpha / \varepsilon^2 \tag{12}$$

In addition, let $\tilde{K}_{ff} = \varepsilon^2 K_{ff}$. Substituting Eq. (12) into Eq. (1) (neglecting the noise terms here) yields

$$\begin{aligned} \ddot{\theta} &= -K_{rr}\theta - \varepsilon^2 K_{rf}\psi - V_{rr}\dot{\theta} - \varepsilon^2 V_{rf}\dot{\psi} + B_r u \\ \varepsilon^2\ddot{\psi} &= -K_{fr}\theta - \tilde{K}_{ff}\psi - V_{fr}\dot{\theta} - \varepsilon^2 V_{ff}\dot{\psi} + B_f u \end{aligned} \tag{13}$$

Letting $\varepsilon = 0$, the slow dynamics equation (denoted by an overbar) is given as

$$\dot{\bar{\boldsymbol{x}}} = \bar{\boldsymbol{A}}\bar{\boldsymbol{x}} + \bar{\boldsymbol{B}}\bar{u} \tag{14}$$

where $\bar{\boldsymbol{x}} = [\bar{\theta}, \dot{\bar{\theta}}]^T$, $\bar{\boldsymbol{B}} = \begin{bmatrix} 0 & B_r \end{bmatrix}$, and

$$\bar{\boldsymbol{A}} = \begin{bmatrix} 0 & 1 \\ -K_{rr} & -V_{rr} \end{bmatrix} \tag{15}$$

For the fast subsystem, a timescale change of $T = t/\varepsilon$ is used, and a new fast variable is defined as

$$\boldsymbol{\phi} = \begin{bmatrix} \psi - \bar{\psi} \\ \varepsilon\dot{\psi} \end{bmatrix} \tag{16}$$

which leads to the fast mode dynamics as

$$d\boldsymbol{\phi} / dT = \tilde{\boldsymbol{A}}\boldsymbol{\phi} + \tilde{\boldsymbol{B}}\tilde{u} \tag{17}$$

where $\tilde{\boldsymbol{B}} = [0, B_f]^T$, the quasi-static solution of the fast mode variable $\bar{\psi}$ is calculated by

$$\bar{\psi} = \tilde{K}_{ff}^{-1}\left(B_f\bar{u} - V_{fr}\dot{\bar{\theta}}\right) \tag{18}$$

and

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} 0 & 1 \\ -\tilde{K}_{ff} & -\varepsilon V_{ff} \end{bmatrix} \tag{19}$$

D. Slow Model Controller

A sliding mode controller (with boundary layer function augmented) is designed for the slow mode as

$$\bar{u} = B_r^{-1}[(\ddot{\bar{\theta}}_d - \lambda\dot{E}) + V_{rr}(\dot{\bar{\theta}}_d - \lambda E)] - \rho\,\mathrm{sat}(S/\beta) \tag{20}$$

where S is the sliding surface defined as

$$S = \dot{E} + \lambda E \tag{21}$$

and β is the thickness of the boundary layer. The tracking error is given by $E = \bar{\theta} - \bar{\theta}_d$. The switching gain is defined as $\rho = \eta\left|\mathrm{sign}(S_0)\right|$, where $S_0 \triangleq S - \beta\,\mathrm{sat}(S/\beta)$, where "sat" denotes the saturation function. The variables $\beta = 0.3$, $\lambda = 3.2$, and $\eta = 0.4$ are selected so that the stability requirement as shown in Appendix A is satisfied and a good performance is achieved. The controller must be tuned carefully, to avoid chattering, which could excite high frequency oscillation. If this occurs, timescale separation is no longer maintained between the slow and fast subsystems.

E. Fast Mode Controller

The fast subsystem is stabilized by the LQR control. Here, the variables in the slow mode are assumed to be constant. The control is given by

$$\tilde{u} = -\tilde{\boldsymbol{K}}\boldsymbol{\phi} \tag{22}$$

where the gain matrix $\tilde{\boldsymbol{K}}$ is chosen to minimize the cost

$$J = \int\left(\boldsymbol{\phi}^T\tilde{\boldsymbol{Q}}\boldsymbol{\phi} + \tilde{u}^T\tilde{R}\tilde{u}\right)dt \tag{23}$$

and after tuning, the following weighting matrices are chosen: $\tilde{\boldsymbol{Q}} = diag([1, 0.1])$ and $\tilde{R} = 70{,}000$.

F. Control Synthesis

The actual control u is calculated through the following steps. First, $\bar{u}$ and $\tilde{u}$ are combined with the consideration of the motor saturation ($-10V \le u \le 10V$),

$$u' = 10\,\mathrm{sat}(\bar{u}/10) + 10\,\mathrm{sat}(\tilde{u}/10) \tag{24}$$

Second, a low pass filter (recommendation from the motor manufacturer) is used to condition the signal so as the motor will not be damaged in case a high frequency command is coming in as

$$u''(s) = [60/(s+60)]u'(s) \tag{25}$$

Finally, the null zone (dead zone) of the motor is considered and the actual control command is derived as

$$u = \begin{cases} -10 & \text{if} \quad u'' < -10 \\ u'' & \text{if} \quad -10 \le u'' \le -0.15 \\ 0 & \text{if} \quad -0.15 < u'' < 0.19 \\ u'' & \text{if} \quad 0.19 \le u'' \le 10 \\ 10 & \text{if} \quad 10 < u'' \end{cases} \tag{26}$$

The motor has a natural dead zone, and it was found that the state estimator's performance improved when the control input reflected this dead zone.

III. Vision Based Tip Point Estimation

By using a camera as the tip point sensing device, a direct, noninvasive measurement can be made. However, a significant delay was found between the actual tip position and measured position, which needs to be compensated for. In this section the camera model used is first explained, followed by the adjustment for distortion in the camera lens and the description of the calibration process. After that how the tip point position is found is discussed.

A. Camera Model

Using the pinhole camera model [18], as illustrated in Fig. 2, the point projection coordinates $\boldsymbol{m} = \begin{bmatrix} u, v \end{bmatrix}^T$, measured in pixels, can be calculated given the position coordinates $\boldsymbol{M} = \begin{bmatrix} X, Y, Z \end{bmatrix}^T$ as

$$s\begin{bmatrix} \boldsymbol{m} \\ 1 \end{bmatrix} = \boldsymbol{A}\begin{bmatrix} \boldsymbol{R} \mid \boldsymbol{t} \end{bmatrix}\begin{bmatrix} \boldsymbol{M} \\ 1 \end{bmatrix} \tag{27}$$

where

$$\boldsymbol{A}=\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{28}$$

Here f_x and f_y are the camera focal length parameters (intrinsic) and $[c_x, c_y]$ is the location of the principle point (intrinsic, the original point on the image frame), whereas the z axis crosses the focal plane as shown in Fig. 2. The arbitrary scale factor s normalizes the left hand side to the right hand side of the equation. For a given camera, these four intrinsic parameters only change when the focus is adjusted. $\boldsymbol{R}\in\Re^{3\times3}$ is the rotation matrix (extrinsic) from the global coordinate to the camera coordinate and $\boldsymbol{t}\in\Re^{3}$ is the translation bias (extrinsic), in which the rotation matrix $\boldsymbol{R}$ is represented in the quaternion form as

$$\boldsymbol{R}=\cos\theta\boldsymbol{I}_{3\times3}+\left[1-\cos(\theta)\right]\hat{\boldsymbol{r}}\hat{\boldsymbol{r}}^{\mathrm{T}}+\sin\theta\tilde{\hat{\boldsymbol{r}}} \tag{29}$$

Here $\boldsymbol{r}=[r_x,r_y,r_z]^T$ (the principle axis of the rotation matrix), $\theta=\|\boldsymbol{r}\|$ (angle of the rotation about the principle axis), $\hat{\boldsymbol{r}}=\boldsymbol{r}/\theta$, and $\tilde{\hat{\boldsymbol{r}}}$ is the corresponding skew symmetric matrix of the vector $\hat{\boldsymbol{r}}$.

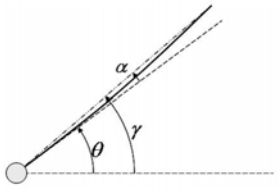

Fig. 1 Flexible beam layout

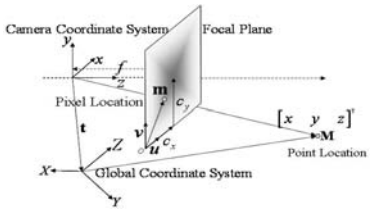

Fig. 2 Pinhole camera model

To solve the distortion problem in the image, resulting from lens deformations, Eq. (27) is modified. A set of intermediate coordinates $\boldsymbol{m}'=[x,y,z]^T$ is defined as

$$\boldsymbol{m}'=\boldsymbol{R}\boldsymbol{M}+\boldsymbol{t} \tag{30}$$

In the case where $z\neq0$, let $x'=x/z$ and $y'=y/z$. Now the distortion can be corrected as [7]

$$\begin{aligned} x'' &= x'\left[1+k_1\left(x'^2+y'^2\right)+k_2\left(x'^2+y'^2\right)^2\right]+2p_1x'y'+p_2\left[\left(x'^2+y'^2\right)+2x'^2\right] \\ y'' &= y'\left[1+k_1\left(x'^2+y'^2\right)+k_2\left(x'^2+y'^2\right)^2\right]+2p_2x'y'+p_1\left[\left(x'^2+y'^2\right)+2y'^2\right] \end{aligned} \tag{31}$$

Here, k_1 and k_2 are radial distortion coefficients, and p_1 and p_2 are tangential distortion coefficients, represented in vector form as $\boldsymbol{k}=[k_1,k_2,p_1,p_2]^T$ (intrinsic parameters). After considering the distortion, the equation used to calculate the camera coordinates is modified as

$$\boldsymbol{m}=\begin{bmatrix} f_x x'' \\ f_y y'' \end{bmatrix}+\begin{bmatrix} c_x \\ c_y \end{bmatrix} \tag{32}$$

Note that when $k_1=k_2=p_1=p_2=0$, $x''=x'$ and $y''=y'$, and the above equations reduce to Eq. (27), where $s=z$.

The process of finding these intrinsic and extrinsic camera properties will be shown in the next section.

B. Camera Calibration

Functions from the Intel® OpenCV library are used to calculate the intrinsic coefficients of a camera. The basic steps involved are shown below. First, the program finds m internal corner points, as shown in Fig. 3 (in this case $m=48$), on a checkerboard pattern in n different orientations (e.g. $n=7$ is used). Without loss of generality, the checkerboard can be assumed to lie on the XY plane of the global coordinate, and the squares are of unit length. Second, the coefficients are obtained through minimizing the residual in the following $2mn$ equation

$$\boldsymbol{J}_{ij}=\boldsymbol{m}_{ij}-\boldsymbol{m}\left(\boldsymbol{A},\boldsymbol{k},\boldsymbol{r}_i,\boldsymbol{t}_i,\boldsymbol{M}_j\right),\ i=1,\ldots,n,\ j=1,\ldots,m \tag{33}$$

$\boldsymbol{m}$ is the calculated point projection coordinates, whereas $\boldsymbol{m}_{ij}$ is the measured coordinates. The vectors $\boldsymbol{r}_i$ and $\boldsymbol{t}_i$ are unique for each orientation of the checkerboard. Because each $\boldsymbol{M}_j$ is known from the checkerboard pattern, and each $\boldsymbol{m}_{ij}$ is measured, the $2mn$ scalar equations have $8+6n$ unknowns (four parameters in $\boldsymbol{A}$ and four distortion parameters. In addition, each image contributes three parameters in $\boldsymbol{r}$ and three parameters in $\boldsymbol{t}$, which gives $8+6n$) (notice $2mn>8+6n$ needs to be satisfied). The calibration program is based off Zhang's work [18] and is a one-time calibration, as $\boldsymbol{A}$ and $\boldsymbol{k}$ are intrinsic properties of the camera.

After finding the intrinsic properties of the camera, the extrinsic properties can be found. Both the camera and global coordinate system are fixed (one-time calibration).

The intrinsic camera properties below are obtained.

Table 1 Intrinsic and Extrinsic Camera Properties

Parameter	Value	Parameter	Value
f_x	783.67	f_y	785.51
c_x	340.66	c_y	246.10
k_1	-0.027	k_2	0.13
p_1	-4.46×10^{-4}	p_2	-3.65×10^{-3}
$\boldsymbol{r}$	$[0.32084,-0.36200,1.5396]^T$		
$\boldsymbol{t}$	$[-0.032443,-0.17023,0.65613]^T$		

C. Target Point Prediction

To find the target point $\boldsymbol{m}$ in the camera projected coordinate, each pixel's red, green, and blue values are tested against a reference color range. Coordinates of matching points are saved, and then averaged to produce the point's projection coordinates. The search range and found pixels are illustrated in Fig. 4. The search range (the lighter dark area) is updated each frame; the smaller size it is, the less computational time it needs in the search algorithm. The global coordinate system is set such that the points on the beam are restricted to move in the XY plane and can be calculated (according to Eq. 32) by

$$x''=(u-c_x)/f_x;\ y''=(v-c_y)/f_y \tag{34}$$

To solve for x' and y', the Nelder-Mead simplex algorithm [19] is used to minimize the residual $x_e^2 + y_e^2$, where x_e and y_e are calculated (based on Eq. 31) by

$$x_e = x'' - x'\left[1 + k_1\left(x'^2 + y'^2\right) + k_2\left(x'^2 + y'^2\right)^2\right] + 2p_1 x' y' + p_2\left[\left(x'^2 + y'^2\right) + 2x'^2\right] \quad (35)$$

and

$$y_e = y'' - y'\left[1 + k_1\left(x'^2 + y'^2\right) + k_2\left(x'^2 + y'^2\right)^2\right] + 2p_2 x' y' + p_1\left[\left(x'^2 + y'^2\right) + 2y'^2\right] \quad (36)$$

Using Eq. (30), the tip point can be found as

$$\boldsymbol{M} = \boldsymbol{R}^{-1}\left(\boldsymbol{m'} - \boldsymbol{t}\right) \quad (37)$$

Because $Z = 0$, z can be found by

$$z = \begin{bmatrix}0 & 0 & 1\end{bmatrix}\boldsymbol{R}^{-1}\boldsymbol{t} \,/\, \left\{\begin{bmatrix}0 & 0 & 1\end{bmatrix}\boldsymbol{R}^{-1}\begin{bmatrix}x' & y' & 1\end{bmatrix}^T\right\} \quad (38)$$

Substituting this result for z into Eq. (37) gives an explicit expression for $[X, Y]^T$.

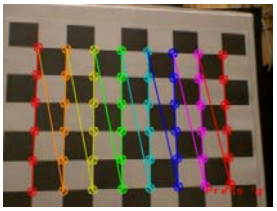

Fig. 3 The check board used for intrinsic parameters' calibration.

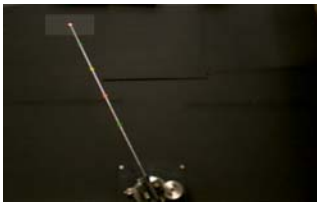

Fig. 4 Target point will be searched in the lighter dark area.

D. Kalman Filtering for Delay Compensation

Although using the camera as a tip sensing device has the advantage of direct measurement, the measurement signal is delayed. The total delay time, determined by comparing the camera measurement with the motor encoder and strain gauge measurements, is roughly 120 ms. The delay is due to the time used in the vision processing and USB signal transmission.

Here a Kalman filter is proposed for the delay compensation and the detailed proof can be found in [20]. The dynamics equation with time delay τ is written as

$$\dot{\hat{\boldsymbol{x}}}_2(t-\tau) = \boldsymbol{A}\hat{\boldsymbol{x}}_2(t-\tau) + \boldsymbol{B}u(t-\tau) + \Gamma\boldsymbol{K}_2\left[y_2 - C_2\hat{\boldsymbol{x}}_2(t-\tau)\right] \quad (39)$$

where the Kalman gain $\boldsymbol{K}_2 = \boldsymbol{P}_2\boldsymbol{C}_2^T R_2^{-1}$, and $\boldsymbol{P}_2$ is the solution of the Ricatti equation

$$\boldsymbol{A}\boldsymbol{P}_2 + \boldsymbol{P}_2\boldsymbol{A}^T - \boldsymbol{P}_2\boldsymbol{C}_2^T R_2^{-1}\boldsymbol{C}_2\boldsymbol{P}_2 + \boldsymbol{Q} = 0 \quad (40)$$

Here R_2 is the covariance of η_2. Because the measurement from the camera is not updated every time step of the simulation, Γ is used as a Boolean variable so as to only apply correction to $\hat{\boldsymbol{x}}_2$ when needed.

To account for the delay, a function $\boldsymbol{g}$ is defined as

$$\dot{\boldsymbol{g}}(t) = \boldsymbol{A}\boldsymbol{g}(t) + \boldsymbol{B}u(t) \quad (41)$$

and the state estimate can now be found as

$$\hat{\boldsymbol{x}}_2(t) = \boldsymbol{g}(t) + e^{\boldsymbol{A}\tau}\left[\hat{\boldsymbol{x}}_2(t-\tau) - \boldsymbol{g}(t-\tau)\right] \quad (42)$$

In Eq. (42), $\hat{\boldsymbol{x}}_2(t-\tau)$ and $\boldsymbol{g}(t)$ are defined above and $\boldsymbol{g}(t-\tau)$ is obtained by a time delay of τ on $\boldsymbol{g}$. Here $\boldsymbol{g}$ is initialized to $[0\ 0\ 0\ 0]^T$, and the Boolean variable for $\hat{\boldsymbol{x}}_2$ is given by

$$\Sigma = \mathrm{e}^{\boldsymbol{A}\tau}\boldsymbol{P}_2\mathrm{e}^{\boldsymbol{A}^T\tau} + \int_0^\tau \mathrm{e}^{\boldsymbol{A}s}\boldsymbol{Q}\mathrm{e}^{\boldsymbol{A}^T s}ds \quad (43)$$

The derivation process is explained more clearly in [20].

The states estimated from the motor measurement $\hat{\boldsymbol{x}}_1$ and the vision measurements $\hat{\boldsymbol{x}}_2$ are then optimally combined in the sense of minimum mean–squared error (MMSE) as [20]

$$\hat{x}_i = \left(\sigma_{i,i}\hat{x}_{1,i} + p_{i,i}\hat{x}_{2,i}\right) / \left(p_{i,i} + \sigma_{i,i}\right),\ i = 1,..,4 \quad (44)$$

where $p_{i,j}$ and $\sigma_{i,j}$ are the i[th] row and j[th] column element of $\boldsymbol{P}_1$ and Σ as given in Eqs. (10) and (43) respectively.

IV. Experiment Validation

A. Experiments Setup

The flexible beam used (manufactured by Quanser®), is part of a complete package. The motor control and sensing devices are connected to the external power module. The power module connects to a data acquisition device, which plugs into the PC via the PCI bus. The camera is a standard web camera, capable of capturing 640x480 frames at 60 Hz and is connected to the PC via a USB port.

B. Experiments Results and Discussion

Two different commanded signals were used in the experiment: a square wave, and a sawtooth wave (or repeating ramp). The input signal is conditioned to limit the maximum velocity of $50\,\mathrm{rad/s}$ and the maximum acceleration of $25\,\mathrm{rad/s^2}$. The results are compared to two controllers: (1) The same control algorithms but the motor position sensors and strain gauge are used instead of the vision feedback, (2) an LQR controller supplied by the manufacturer of the flexible beam where the motor sensors and strain gauge are used. The feedback gain used in the LQR control is $\boldsymbol{k} = \begin{bmatrix}14.14, -62.81, 2.289, 0.4993\end{bmatrix}^T$.

Figure 5 illustrates the response to a square wave. It can clearly be seen that the state estimator compensates for the delay from the camera signal. In Fig. 6, the control commands of the proposed method from both the slow and fast controllers are shown, along with the resulting combined signal. In Fig. 7 a comparison is made between the different control methods in response to a square wave. All three controllers have a similar initial response to the step input. However, the proposed controller has the shortest settling time and smallest steady state error. Despite the fact that position data from the camera is delayed, the controller outperforms the standard LQR controller, which uses instantaneous position data from the motor and strain

gauge. In addition, the control command (Fig. 8) is much smoother (a big advantage), and drops off to nearly 0V after 1 second.

The comparison of the three control methods in response to the sawtooth wave is shown in Figs. 9 and 10. The SMC with strain gauge feedback never settles down on the command, though it does hover around it (Fig. 9). The LQR controller creates a smooth curve, but lags behind the command and has a big steady state error. As shown in Fig. 10, the control signal of the proposed method is less oscillating than the other two methods.

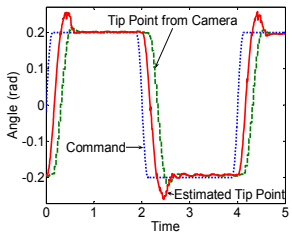

Fig. 5 Tip point estimation (square wave)

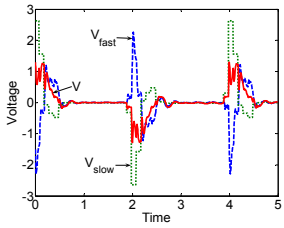

Fig. 6 Control commands for the slow and fast modes (vision based - square wave)

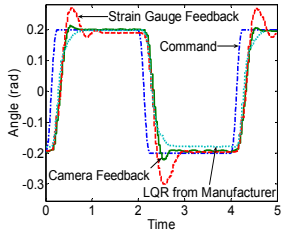

Fig. 7 Tip point tracking comparison (square wave)

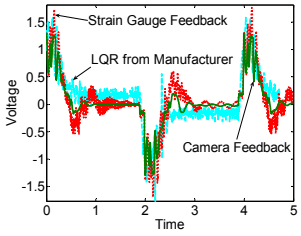

Fig. 8 Control commands comparison (square wave)

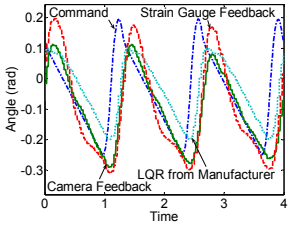

Fig. 9 Tip point tracking comparison (sawtooth wave)

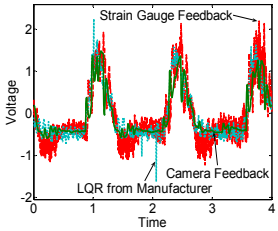

Fig. 10 Control commands comparison (sawtooth wave)

V. Conclusion

Using flexible structures offers many advantages, stemming from their reduced weight. However, large settling times and tip point vibration may decrease its productivity. In this paper, a vision based two-timescale controller was developed to track a desired tip position while mitigate oscillations in the single-link flexible beam. The controller demonstrated through experiments is proved to exceed existing control schemes. In the case of a step input, it quickly reached its target with minimal oscillations and steady state error. In the case of the ramp input, it is able to track the moving signal with no oscillations.

References

[1] H. Kojima, and W. Singhose, "Adaptive deflection-limiting control for slewing flexible space structures," *J. Guid. Control Dyn*., vol. 30, no. 1, pp. 61-67, Jan. – Feb. 2007

[2] Y. Liu, and D. Sun, "Stabilizing a flexible beam handled by two manipulators via PD feedback," *IEEE Trans. Autom. Control*, vol. 45, no. 11, pp. 2159-2164, Nov. 2000.

[3] B. K. Kim, S. Park, W. K. Chung, and Y. Youm, "Robust controller design for PTP motion of vertical XY positioning systems with a flexible beam," *IEEE/ASME Trans. on Mechatron.*, vol. 8, no. 1, pp. 99-110, Mar. 2003.

[4] B. Yang, A. Calise, and J. Craig, "Adaptive output feedback control of a flexible base manipulator," *J. Guid. Control Dyn*., vol. 30, no. 4, pp. 1068-1080, July-Aug. 2007

[5] V. Etxebarria, A. Sanz, and I. Lizarraga, "Control of a lightweight flexible robotic arm using sliding modes," *Int. J. Adv. Robot. Syst.,* vol. 2, no. 2, pp. 103-110, 2005.

[6] M. Weng, X. Lu, and D. L. Trumper, "Vibration control of flexible beams using sensor averaging and actuator averaging methods," *IEEE Trans. on Control Syst. Technol.*, vol. 10, no. 4, pp. 568-577, July 2002.

[7] D. Brown, "Close-range camera calibration," *Photogrammetric Eng.*, vol. 37, pp. 855-866, 1971.

[8] M. Stieber, M. McKay, G. Vukovich, and E. Petriu, "Vision-based sensing and control for space robotics applications," *IEEE Trans. on Inst. and Meas.*, vol. 48, no. 4, pp. 807-812, Aug 1999.

[9] P. Tang, H. Wang, and S. Lu, "A vision-based position control system for a one-link flexible arm," *Proc. of the 1990 IEEE Int. Workshop on Intell Motion Control*, vol. 2, pp. 523-528, Aug. 20-22, 1990

[10] R. Cannon, and E. Schmitz, "Initial experiments on the end-point control of a flexible one-link robot," *the Int. J. Robot. Res.*, vol. 3, no. 3, pp. 62-75, 1984.

[11] N. Chalhoub, and A. Ulsoy, "Control of a flexible robot arm: experimental and theoretical results," *J. Dyn. Syst., Meas. and Control*, vol. 109, pp. 299-309, Dec 1987

[12] A. Pisoni, C. Santolini, D. Hauf, and S. Dubowsky, "Displacements in a Vibrating Body by Strain Gauge Measurements," *Pro. of the 13th Int. Conf. on Modal Anal.*, Nashville, TN, 1995.

[13] O. Kaynak, and A. Denker, "Discrete-time sliding mode control in the presence of system uncertainty," *Int. J. Control*, vol. 57, no. 5, pp. 1177-1189, 1993.

[14] P. Korondi, H. Hashimoto, and V. Utkin, "Direct torsion control of flexible shaft in an observer-based discrete-time sliding mode," *IEEE Trans. Indu Electron.*, vol. 45, no. 2, pp. 291-296, April 1998

[15] S. Madhavan, and S. Singh, "Sliding mode end point trajectory control of a two link elastic manipulator," *Proc. of the 31st IEEE Int. Conf. on Decision and Control*, pp. 305-310, Dec. 16-18 1992

[16] R. Garrido, A. Soria, P. Castillo, I. Vasquez, "A visual servoing architecture for controlling electromechanical systems," *Proc. of the 2001 IEEE Int. Conf. on Control Appl.*, pp. 35-40, 2001.

[17] H. Khalil, *Nonlinear Systems*, 3rd Edition, Prentice Hall, New Jersey, 2002, pp. 423-459.

[18] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. on Pattern Anal. and Mach. Intell.*, vol. 22, no. 11, pp. 1330-1334, Nov. 2000.

[19] S. Hirai, http://www.ritsumei.ac.jp/se/~hirai/edu/2003/algorithm/index-e.html, Document Accessed: Oct. 12, 2007.

[20] A. Roberts, "State estimation when some measurements are delayed," *IMA J. Math. Control & Inf.*, vol. 3, no. 4, pp. 299-310, 1986.

[21] J. E. Slotine and W. Li, *Applied Nonlinear Control*. New Jersey: Prentice Hall, 1990, pp. 267-307.