2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

A Vision-Based Endpoint Trajectory and Vibration Control for Flexible Manipulators

Xin Jiang, Atsushi Konno, *Member*, *IEEE*, and Masaru Uchiyama, *Member*, *IEEE*

***Abstract*— This paper addresses a vision-based endpoint trajectory and vibration control approach for flexible manipulators. In this approach, both the trajectory and vibration control are implemented by using an endpoint camera. In order to estimate link vibrations from the visual information, Kalman filter is used. To make the endpoint to follow a specified trajectory, image interpolation approach is introduced. The experimental results prove the effectiveness of the proposed control strategy.**

I. INTRODUCTION

Present space manipulators such as SRMS (Space Shuttle Remote Manipulator System) and SSRMS (Space Station RMS) have structural flexibility in their arms and joints. Due to the lack of rigidity, flexible manipulators severely suffer from the structural vibration and inaccuracy in end-effector positioning. Therefore, astronauts operate the space manipulators very slowly in order to avoid exciting structural vibrations. If some operations in space applications are automated, the load of astronauts can be drastically reduced. In the automated task operation, it is necessary to consider the structural flexibility of the manipulator.

One possible solution to improve the positioning accuracy is the usage of camera. It is a feasible solution because most of space manipulators are equipped with an endpoint camera. There are two approaches for the endpoint control of rigid manipulators using vision: Position Based Visual Servoing (PBVS) and Image Based Visual Servoing (IBVS). In Position Based Visual Servoing, the visual information is used to estimate the relative pose between the end-effector and the target. The control is aimed to reduce the pose error. This approach necessitates precise robot model and camera model. To the contrary, in Image Based Visual Servoing, the visual information is directly fed back for control. The control objective is to reduce the error defined in image plane. In general, it is difficult to apply the position-based approach to a flexible manipulator, because the structural deflection cannot be directly measured but can be estimated from the sensor information such as strain at each link. Inaccuracy in the estimation of the structural deflection may bring a significant error in kinematics solution. It may be rather easy to apply image-based approach to a flexible manipulator, because this kind of approach is robust to the error in robot kinematics and camera model. Even if a rough end-effector Jacobian matrix and imprecisely calibrated camera are used in control, the convergence of control can still be guaranteed in the neighbourhood of the target. This property provides flexible manipulators a simple way to compensate the kinematics error due to link's bending. Therefore, an image-based visual feedback is discussed in this paper.

The image-based approach was originally developed for rigid manipulators, and hence if it is directly applied to flexible manipulators, the approach may bring instability in the control, since the visual image vibrates due to the structural flexibility of the manipulator. Therefore, an image-based visual feedback control for flexible manipulator was proposed [1], in which only the low frequency changes in image were used for endpoint control. The link vibration was suppressed using links' strain in [1].

Except the approach of input shaping, most of vibration control strategies proposed so far assume that the structural vibration is directly measured by sensors such as strain gauges or accelerometers. However, present space manipulators such as SRMS and SSRMS are not equipped with such kinds of sensors. Therefore a vibration suppression control using an endpoint camera is also discussed in this paper.

Some researchers have tried to use a camera to estimate the structural vibration. Bascetta fused the visual information into the strain information using a Kalman filter. The less noisy measurement from camera helps improve the quality of deflection measurement. However no experimental results only using camera to damp out the vibration were provided in [2].

As for the endpoint control, the image-based approach controls the end-effector following the trajectory minimizing the error defined in image plane and hence sometimes the generated trajectory is not appropriate for a given task. For this problem an image interpolation based solution has been proposed in [3], which specifies the end-effector's trajectory by using images of intermediate points. This approach is applied to the endpoint trajectory control of a flexible manipulator in this paper. The proposed approach is verified by a peg insertion task.

The paper is organized as follows. Sections I and II present an introduction and the dynamic model of a 3D flexible manipulators, respectively. A vibration suppression control using vision is proposed in section III. Image interpolation is integrated in order to specify the endpoint trajectory. The detailed implementation is described in IV, which is followed by experimental verification in section V. The paper ends with concluding remarks in section VI.

This work was partly supported by NTT DoCoMo, Inc.

The authors are with the Department of Aerospace Engineering, Tohoku University, Aoba-yama 6-6-01, Sendai 980-8579, Japan (e-mail: {jiangxin,konno,uchiyama}@space.mech.tohoku.ac.jp)

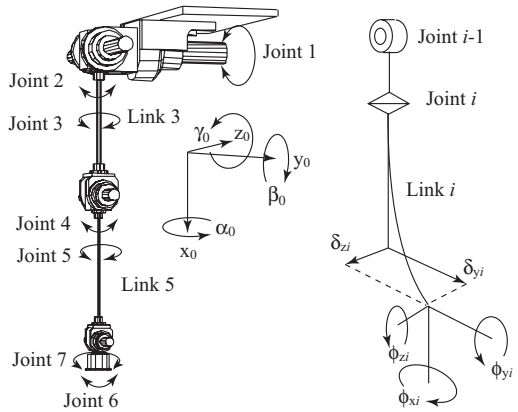


Fig. 1. Overview of a flexible manipulator and deflection definition.

II. Dynamic model of a flexible robot

The 3D flexible-link manipulator shown in Fig. 1 is used in this study. The flexible manipulator is modelled by lumped masses and massless springs. The masses are assumed to be concentrated on the shoulder, elbow, wrist, and end-effector. They are considered to be connected with massless flexible links. The dynamics equation can be expressed as:

$$\begin{bmatrix}\tau\\0\end{bmatrix} = \begin{bmatrix}M_{11}(\theta,e) & M_{12}(\theta,e)\\ M_{21}(\theta,e) & M_{22}(\theta,e)\end{bmatrix}\begin{bmatrix}\ddot{\theta}\\ \ddot{e}\end{bmatrix} + \begin{bmatrix}h_1(\theta,\dot{\theta},e,\dot{e})\\ h_2(\theta,\dot{\theta},e,\dot{e})\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & K_{22}\end{bmatrix}\begin{bmatrix}\theta\\ e\end{bmatrix} + \begin{bmatrix}g_1(\theta,e)\\ g_2(\theta,e)\end{bmatrix}, \quad (1)$$

where $\theta = [\theta_1 \ \cdots \ \theta_n]^T$ is the joint angle vector and $e = [e_1, \ \cdots e_m]^T$ is the deflection variable vector, τ stands for the input joint torque vector, M_{11}, M_{12}, M_{21} and M_{22} are inertia matrices, h_1 , h_2 are the centrifugal and Coriolis force vectors, K_{22} represents the stiffness matrix and g_1 , g_2 are gravity vectors. A set of velocity controlled actuators is used in the system, and thus the actual inputs to the system is formed as a set of angle velocity reference for each joint.

This paper deals with relatively slow motions, and assumes that the centrifugal and Coriolis terms can be neglected. The elastic vibration is excited around the equilibrium state of each joint configuration, where the bending of links takes balance with gravity effect. Referring to (1), the following equilibrium conditions can be derived:

$$\begin{bmatrix}0 & 0\\ 0 & K_{22}\end{bmatrix}\begin{bmatrix}\theta_0\\ e_0\end{bmatrix} + \begin{bmatrix}g_1(\theta_0)\\ g_2(\theta_0)\end{bmatrix} = \begin{bmatrix}\tau_0\\ 0\end{bmatrix}, \quad (2)$$

where θ_0 is a given joint configuration, e_0 is the static deflection, and τ_0 is the torque to maintain balance. The influence of deflection on gravity terms is ignored. Δe and $\Delta\tau$ are defined to represent the deviation of corresponding variables from their static values as follows:

$$\Delta e = e - e_0 = e + K_{22}^{-1} g_2(\theta), \quad (3)$$

$$\Delta\tau = \tau - g_1(\theta). \quad (4)$$

With these variables, a linearized model is derived as:

$$\begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}\begin{bmatrix}\ddot{\theta}\\ \Delta\ddot{e}\end{bmatrix} + \begin{bmatrix}0 & 0\\ 0 & K_{22}\end{bmatrix}\begin{bmatrix}\theta\\ \Delta e\end{bmatrix} = \begin{bmatrix}\Delta\tau\\ 0\end{bmatrix}, \quad (5)$$

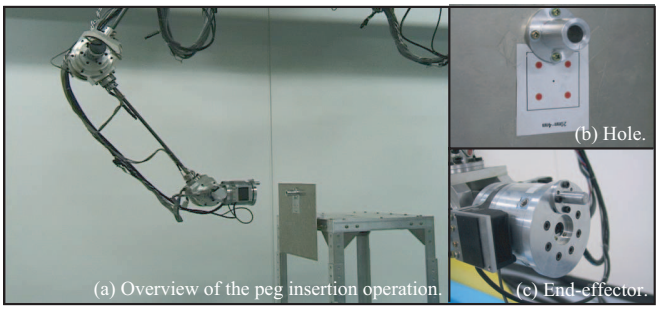


Fig. 2. Experimental setup.

where Δe is computed from (3). It should be noticed that the lower part of (5) can be thought to dominate the behaviour of vibration. For motions in free space, it can be proved that the deflection variables are not independent. The independent variables are chosen to be the bending deflection of $e = \begin{bmatrix}\delta_{y3} & \delta_{z3} & \delta_{y5} & \delta_{z5}\end{bmatrix}$ as shown in Fig. 1.

For the control of a task level operation, a composite strategy is used, in which the whole dynamics is divided into two subsystems: the rigid motion of task level as a slow subsystem, and the elastic vibration as a fast subsystem. The overall controller is simply defined as the sum of the sub-controllers independently designed for each subsystem:

$$\dot{\theta}_c = \dot{\theta}_r + \dot{\theta}_e, \quad (6)$$

where $\dot{\theta}_r$ and $\dot{\theta}_e$ refer to the control signals contributing to rigid motion and elastic one respectively.

III. A vision-based vibration suppression control

In the case of three dimensional flexible manipulators, a simple strain feedback with fixed gain may bring instability, since the relation between actuated joints and vibration changes depending upon the posture of the arm [4]. Therefore, a configuration dependent strain feedback is proposed to damp out the vibration of the links [4].

Based on the configuration dependent strain feedback control, a vision-based vibration control is derived in this section. The lower component of (5) is rewritten here:

$$M_{22}\Delta\ddot{e} + K_{22}\Delta e = -M_{21}\ddot{\theta}. \quad (7)$$

(7) describes elastic motion of the flexible manipulator. In (7), joint acceleration $\ddot{\theta}$ affects the vibration system as if it were the input to the system. Therefore, if the damping effect for vibration is realized by joint acceleration $\ddot{\theta}$, the vibration is well suppressed. Assuming ideal velocity controlled motors in which the motors strictly follow the given velocity command, the following vibration damping command is considered:

$$\dot{\theta}_e = M_{21}^{+} M_{22} K_e \Delta e, \quad (8)$$

where "+" refers to the pseudo inverse, and K_e is a diagonal gain matrix. Time derivative of (8) is given by:

$$\ddot{\theta}_e = M_{21}^{+} M_{22} K_e \Delta\dot{e}. \quad (9)$$

Note that since the posture of the manipulator does not drastically change, the time derivative of the inertia matrices is approximated to be zero in (9). Substituting (9) into (7), the following second order damping vibration system is obtained:

$$\Delta\ddot{e} + K_e\Delta\dot{e} + M_{22}^{-1}K_{22}\Delta e = 0. \tag{10}$$

Modal transformation $\Delta e = \Phi\Delta e^*$ is applied to (10) as follows:

$$\Delta\ddot{e}^* + \Phi^{-1}K_e\Phi\Delta\dot{e}^* + \Phi^{-1}M_{22}^{-1}K_{22}\Phi\Delta e^* = 0, \tag{11}$$

where Φ and Δe^* are the modal matrix and modal coordinate, respectively. If the gain matrix K_e is given as $K_e = K_e I$ (I is the unit matrix), (11) becomes as follows:

$$\Delta\ddot{e}^* + K_e I\Delta\dot{e}^* + \Omega\Delta e^* = 0, \tag{12}$$

where Ω is a diagonal matrix whose diagonal elements indicate the frequency of the corresponding mode. (12) denotes the mode transformed damping vibration system.

In [4], the link deflection e is calculated from the strain. In this section, a vision-based vibration control is discussed assuming that no other sensors except for an endpoint camera are available to detect the structural vibration. If no sensors are available to directly measure the deflection, an observer will be needed to provide an estimation of deflection for vibration control. A Discrete Kalman filer is used for estimation. According to (7), the state space model of vibration can be described as:

$$\begin{bmatrix}\Delta\dot{e}\\ \Delta\ddot{e}\end{bmatrix} = \begin{bmatrix}0 & I\\ M_{22}^{-1}K_{22} & 0\end{bmatrix}\begin{bmatrix}\Delta e\\ \Delta\dot{e}\end{bmatrix} + \begin{bmatrix}0\\ -M_{22}^{-1}M_{21}\end{bmatrix}\ddot{\theta}$$
$$\dot{x} = Ax + Bu, \tag{13}$$

where $x = [\Delta e, \Delta\dot{e}]^T$.

The measurement for the observer comes from the reading of the endpoint camera. For simplicity, it is assumed that the feature points are stationary. In the experimentation presented in Section V, four marks are attached to an object as feature points, as shown in Fig. 2.

Both the movement of the joints and vibration affect the velocity of the feature points in the image plane. Therefore, the velocity of the feature points expressed in difference formula, $\Delta\xi$, is divided into two components: low frequency component $\Delta\xi_{low}$ and high frequency component $\Delta\xi_{high}$. It is assumed that the movement of the joints mainly affects $\Delta\xi_{low}$, while the vibration affects $\Delta\xi_{high}$. Hence $\Delta\xi_{low}$ is used for the endpoint control, and $\Delta\xi_{high}$ is used for the vibration suppression control. The high frequency component is expressed as follows:

$$\Delta\xi_{high} = \begin{bmatrix}J_{image}J_e & 0\end{bmatrix}\begin{bmatrix}\Delta e\\ \Delta\dot{e}\end{bmatrix}$$
$$z = Cx, \tag{14}$$

where J_e refers to end-effector Jacobian with respect to link deflection, while J_{image} refers to the Image Jacobian matrix, which relates the velocity of the end-effector with the velocity of the feature points in the image plane [5].

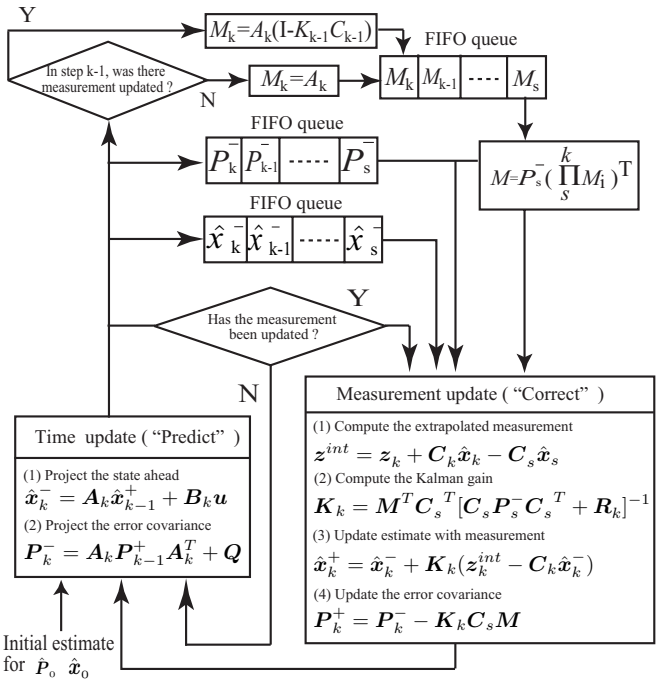


Fig. 3. A two-time-scale discrete Kalman filter with a delay compensator.

Before applying a discrete Kalman filter to the system expressed by (13) and (14), it is necessary to consider some technical problems. One problem is the difference between the output measurement rate and the servo rate. Since a NTSC standard camera is employed in this experiment, z in (14) is updated at the NTSC camera frame rate (30 Hz). On the other hand, in order to guarantee the stability and control performance, the servo rate in the experiments is set at 128 Hz, which is approximately four times faster than the camera frame rate. This difference brings a difficulty in the implementation of the discrete Kalman filter.

The other problem is a delay in the output measurement expressed by (14). It takes approximately one video frame to capture a camera image in a frame memory, and takes approximately 10 $\sim$ 20 ms for image processing. These delays cause a phase-lag between the real state and estimated state. The phase-lag may bring instability into the vibration control system.

In order to overcome the above two problems in the implementation of the standard discrete Kalman filter, the Kalman filter is modified and a delay compensation is supplementarily used as shown in Fig. 3.

The standard discrete Kalman filter executes (1), (2) in the *Time update* block and (2), (3), (4) in the *Measurement update* block in Fig. 3 at the same time in this order. However, z_k is updated at the camera frame rate (30 Hz), while estimation is expected to be done at the servo rate (128 Hz). Therefore, the processes in the discrete Kalman filter are split into two time-scales: the *Time update* time-scale and the *Measurement update* time-scale.

In the proposed two-time-scale discrete Kalman filter

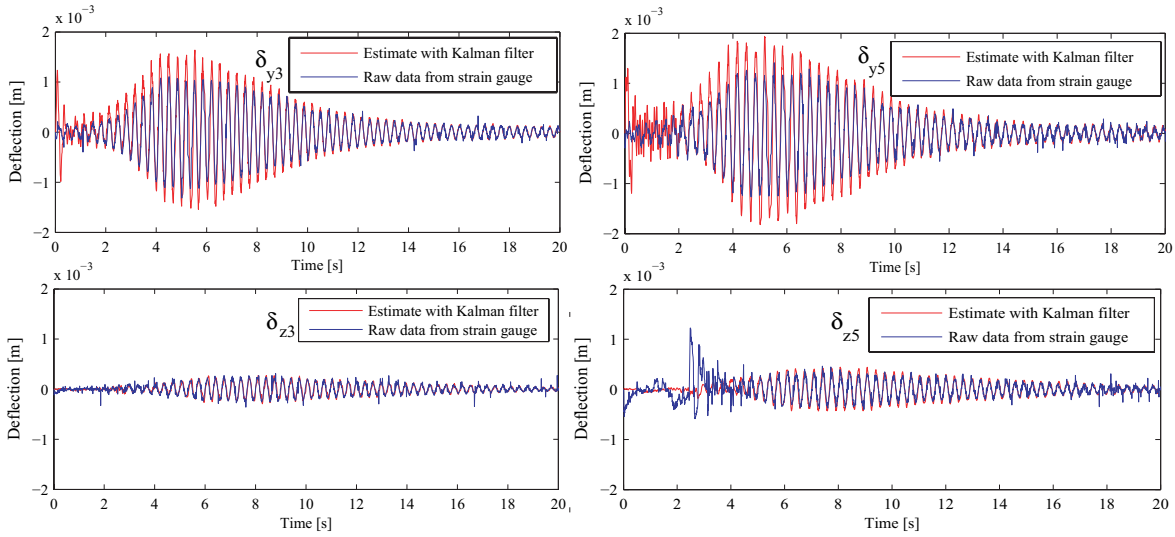


Fig. 4. Comparison between the vibration measured by strain gauges (blue) and vibration estimated from the the endpoint camera view (red).

illustrated in Fig. 3, a flag is introduced to show whether the measured output (z in (14)) is updated or not. The *Time update* block in Fig. 3 is executed at the servo rate (128 Hz). When the measured output is updated, the *Measurement update* block is executed to refine the estimation. The proposed two-time-scale discrete Kalman filter is expected to be robust against the visual problems such as occlusion, because even if the vision system fails to compute z_k, the *Time update* block does not stop estimating the state variables.

In order to compensate the delay involved in the output measurement, the approach proposed in [6] is applied to the estimation. As shown in Fig. 3, the delay compensator extrapolates the measured output to the present time using past and present estimates of the Kalman filter. The delay is estimated from the preliminary experiments to be five servo sampling times. Therefore, five past data are memorized in each FIFO queue. Larsen et al. assumed the standard discrete Kalman filter [6]. In this paper, the delay compensator proposed in [6] is applied to the two-time-scale Kalman filter as illustrated in Fig. 3.

In the two-time-scale Kalman filter illustrated in Fig. 3, $\hat{x}_k^-$ represents the prior estimate. $\hat{x}_k^+$ represents the posteriori estimate incorporating the delayed measurement z_k. z_k is obtained through a high-pass filter to eliminate the low frequency robot motion. The cut-off frequency of the high-pass filter is 0.5 Hz. Due to the delay, the measurement obtained at kth iteration is actually taken at sth iteration ($k > s$). As proposed in [6], in order to incorporate the delayed measurement, an interpolated output $z^{int} = z_k + C_k\hat{x}_k^- - C_s\hat{x}_s^-$ is defined. $\hat{x}_k^-$, $\hat{x}_s^-$ represent the prior estimates made at kth iteration and sth iteration respectively. This interpolated output enables the Kalman filter to treat the delayed measurement. Q and R are the diagonal covariance matrices of process noise and measurement noise, whose diagonal components are 11 μm^2 and 0.4 pixel^2, respectively.

An experiment is carried out to evaluate the performance of the Kalman filter. In this experiment, the end-effector is commanded to move horizontally. The vibration is controlled with a very small gain to clearly compare the estimated value using Kalman filter with the measured value avoiding danger. The link deflections δ_{y3}, δ_{y5}, δ_{z3}, and δ_{z5} (see Fig. 1) estimated using the Kalman filter is compared with the deflections calculated from the strain gauge signal as shown in Fig. 4. The strain data is high-pass filtered to eliminate the gravitational deflection with a cut-off frequency of 1.0 Hz. It is possible to say from Fig. 4 that the estimation using the Kalman filter shows satisfactory accuracy. Note that the natural frequency at the experimented configuration of the arm is about 3 Hz for the fist mode and 11 Hz for the second mode. That is to say the sampling speed of camera as 30 Hz is fast enough according to sampling theory. The vibration suppression control (8) is experimented using the endpoint camera image and Kalman filer instead of using a strain gauge. The results are presented in Section V.

IV. Specifying the End-Effector Trajectory Using Image Interpolation

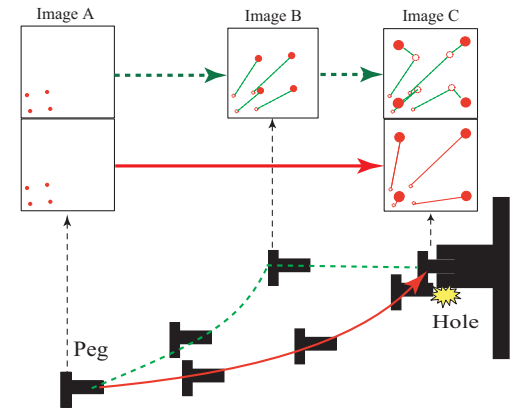
In (6), the component $\dot{\theta}_r$ contributes for the endpoint control. In this paper, an image-based visual servo is designed as:

$$\dot{\theta}_r = K_I J_\theta^+ J_{image}^+ \Delta\xi_{low}, \tag{15}$$

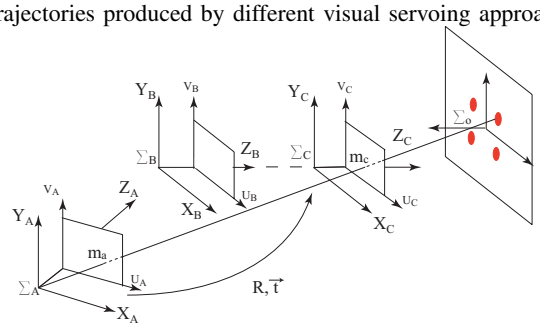
$$\Delta\xi_{low} = S_{low}(\xi_d - \xi), \tag{16}$$

where K_I denotes the gain for visual servo. As described in Section III, only the low frequency component of the feature points $\Delta\xi_{low}$ is used for the visual servo.

The image-based visual servo described in (15) controls the robot manipulator to decrease $\Delta\xi_{low}$ linearly. Therefore, there is no guarantee that the resultant endpoint trajectory is adequate for a given task. Fig. 5(a) shows an example in which the resultant endpoint trajectory (the solid-red curve) is not adequate. As a resolution to this problem, Mezouar et al. proposed an extension of the image-based approach [3], in which a series of images is given as the reference.



(a) Trajectories produced by different visual servoing approaches.



(b) Geometry of views.

Fig. 5. Parts mating guided by interpolated images.

Each image in the series represents a successive frame seen by the camera when the end-effector follows the specified trajectory.

The method proposed in [3] is simplified and applied to the particular peg-in-hole problem here to show its potential as a solution for the motion planning problem of visually guided flexible robots. The method proposed in [3] uses a sequence of real images along the endpoint trajectory, while the method proposed in this section uses only the final image. The feature points in the final image are extracted, and intermediate feature points along the endpoint trajectory are calculated from the final feature points.

For the peg-in-hole task, the natural strategy is to align the approaching axis of the peg with the axis of the hole. Therefore an intuitive solution is to insert intermediate reference feature points between the initial and final reference feature points so that the peg squarely faces the hole before peg-insertion (dashed-green curve in Fig. 5(a)). In Fig. 5(a), image A refers to the start of operation, and image C refers to the end. An image B is inserted as an additional reference before the peg reaches the target. The reference feature point ξ_d in (15) at every sampling period is given by interpolating between images A and B or between images B and C. The feature points in the image B are produced from the feature points in the image C so that the axis of the peg is aligned with the axis of the hole. Image A is the initial camera view image before the operation starts. Note that only the image C is needed before the experiment because the reference feature points at any moment in the operation can be produced from the feature points in the image C and the feature points taken in the initial position. As shown in Fig. 5(b), the camera coordinate frames corresponding to image A, B, C are denoted as Σ_A, Σ_B, Σ_C and the plane containing marks is denoted as Σ_O.

When a feature on a plane is observed by a moving camera, its coordinate with respect to camera frame at a different instant can be related by a simple representation [7]. Let the coordinate of feature j with respect to frame A and C be ${}^A p_j = \begin{bmatrix} {}^A X_j & {}^A Y_j & {}^A Z_j \end{bmatrix}^T$, ${}^C p_j = \begin{bmatrix} {}^C X_j & {}^C Y_j & {}^C Z_j \end{bmatrix}^T$, then the relationship is expressed by:

$$ {}^A p_j = {}^A R_C {}^C p_j + {}^A t_C, \tag{17} $$

where ${}^A R_C$ and ${}^A t_c$ are the rotation matrix and the translation vector from frame A to C, respectively. Considering that the feature is located in the plane, the following relationship is obtained:

$$ {}^C n \cdot {}^C p_j = {}^C n_x {}^C X_j + {}^C n_y {}^C Y_j + {}^C n_z {}^C Z_j = {}^C d, \tag{18} $$

where ${}^C n = \begin{bmatrix} {}^C n_x & {}^C n_y & {}^C n_z \end{bmatrix}^T$ is a unit vector defined with respect to frame C, which is perpendicular to the plane Σ_O. ${}^C d$ is the distance between Σ_C and Σ_O. Using (17) and (18), ${}^A p_j$ is calculated as:

$$ {}^A p_j = \left({}^A R_C + \frac{{}^A t_C {}^C n^T}{{}^C d} \right) {}^C p_j = G_0 {}^C p_j. \tag{19} $$

Let the frames A, B, and C to be re-indexed as 0, 1, and 2. The transform matrix from frame A to C and from frame B to C are defined as G_0 and G_1, respectively. In order to interpolate between the images A and B or between images B and C, interpolation of the transform matrices between G_0 and G_1, and between G_1 and I is considered. Consider that the camera is modelled by a simple pin-hole, and in addition a set of known coplanar image features are employed. In this condition, the rough relative position between the frames can be calculated as in [5]. It makes the calculation of G_i matrices possible. With the calculated G_i matrices, the interpolation between two transform matrices is given as follows [3]:

$$ G(\tau) = (1-\tau)\phi_{i-1} + \tau\phi_i + (G_{i-1} - \phi_{i-1})\Gamma \tag{20} $$

$$ \Gamma(\theta_i, \tau) = e^{([\theta_i]_\times \tau)} \tag{21} $$

$$ \phi_i = \frac{{}^i t_C {}^C n^T}{{}^C d} \tag{22} $$

where t_i is the time when camera reaches the corresponding frame i. τ and $[\theta_i]_\times$ are defined as $\tau = \frac{t - t_{i-1}}{t_i - t_{i-1}}$ and $[\theta_i]_\times = \log({}^{i-1}R_C^T {}^i R_C)$, respectively. ${}^i t_C$ and ${}^i R_C$ are the translation vector and rotation matrix from frame i to frame C. In the above equations, operations $e^{[]}$, $\log()$ are defined in Lie algebra. Their calculations can be referenced from [3]. The position of the feature j at the instant k can be calculated from ${}^C p_j$ as follows:

$$ {}^k p_j(\tau) = G(\tau) {}^C p_j. \tag{23} $$

${}^k p_j$ is the relative three dimensional position vector with respect to the camera coordinate frame. Using pin-hole camera model, a reference feature points in the image plane ξ_d used in (15) can be produced from ${}^k p_j$.

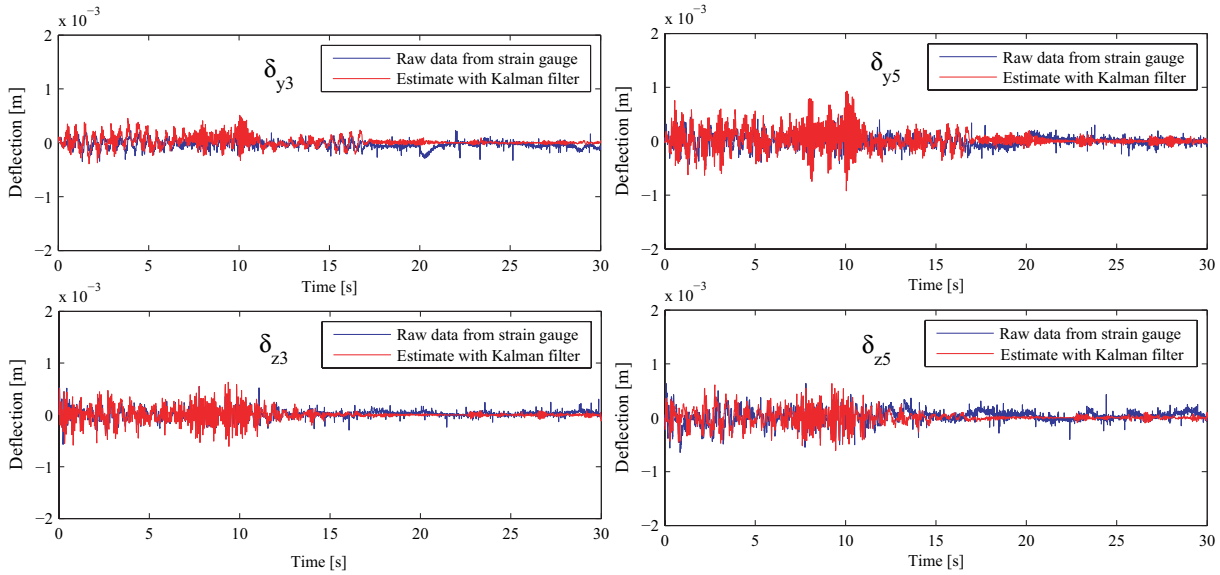


Fig. 6. Estimated and actual deflection during the vision guided peg-in-hole experiment

V. Verification Experiment

Combining the vision-based vibration suppression control discussed in III and endpoint trajectory control using image interpolation in IV, a verification experiment is carried out. Due to the limitation in the narrow view range of camera, it is still difficult to perform the whole of positioning task only with a camera. Therefore before the experiment starts, the end-effector is positioned to a stable state with strain gauge feedback. In this state the vibration has been damped very well and the features are already detected by the camera. From this position, a peg-in-hole operation is performed which is characterized in the fact that both the trajectory following and vibration damping are implemented using camera. As shown in Fig. 6, the vision-based vibration suppression works very well. With image interpolation scheme applied, the peg squarely faces the hole before contact, which makes the insertion easier as shown in Fig. 7. In this experiment the length of the trajectory is defined as 20 s with the interval between image A, B, and C is defined as 10 s.

VI. Conclusion

In this paper, a vision-based endpoint trajectory and vibration control scheme is proposed for flexible manipulators. The vibration suppression control is designed using visual information. Four marks are used as the visual target for the visual servoing. In order to specify the endpoint trajectory, a reference visual targets interpolation approach is introduced.

A discrete Kalman filter is used to estimate the deflection and vibration of the links using the endpoint camera. In order to overcome the difference between the NTSC video frame rate (30 Hz) and the servo rate (128 Hz), a two-time-scale discrete Kalman filter is proposed.

In the control of flexible manipulators including real space manipulators, robot motion is designed slower than the vibration of natural frequency of links, in order to avoid the resonance. The proposed scheme utilizes the difference in time-scale between the robot motion and the natural frequency of the links.

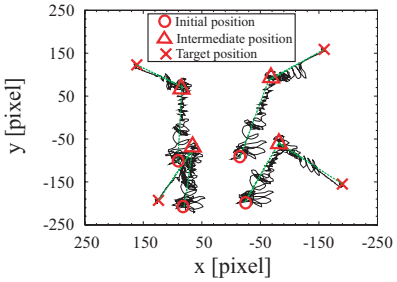


Fig. 7. Feature trajectory with image interpolation

References

[1] X. Jiang, A. Konno, and M. Uchiyama, "Visual servoing experiment using a 3d flexible-link manipulator," in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2006, pp. 1224–1229.

[2] P. R. Luca Bascetta, "End-point vibration sensing of planar flexible manipulators through visual servoing," *Mechatronics*, vol. 16, pp. 221–232, 2006.

[3] Y. Mezouar, A. Remazeilles, R. Gros, and F. Chaumette, "Image interpolation for image based control under large displacement," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, 2002, pp. 3787–3794.

[4] A. Konno and M. Uchiyama, "Vibration suppression control of spatial flexible manipulator," *Control Engineering Practice, A Journal of IFAC*, vol. 3, no. 9, pp. 1315–1321, 1995.

[5] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator control with image-based visual servo," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, 1991, pp. 2267–2272.

[6] T. Larsen, N. Poulsen, N. Andersen, and O. Ravn, "Incorporation of time delayed measurements in a discrete-time kalman filter," in *proceedings for CDC '98, Tampa, Floridac*, 1998.

[7] O. Faugeras and F. Lustman, "Motion and structure from motion in a piecewise planar environment," *Int. Journal of Pattern Recognition and Artifical Intelligence*, vol. 2, no. 3, pp. 485–508, 1988.