2017 IEEE International Conference on Robotics and Automation (ICRA)
Singapore, May 29 - June 3, 2017

Vision-Based Model Predictive Control for Within-Hand Precision Manipulation with Underactuated Grippers

Berk Calli, *Member, IEEE*, and Aaron M. Dollar, *Senior Member, IEEE*

***Abstract*— Precision manipulation with underactuated hands is a challenging problem due to difficulties in obtaining precise gripper, object and contact models. Using vision feedback provides a degree of robustness to modeling inaccuracies, but conventional visual servoing schemes may suffer from performance degradation if inaccuracies are large and/or unmodeled phenomena (e.g. friction) have significant effect on the system. In this paper, we propose the use of Model Predictive Control (MPC) framework within a visual servoing scheme to achieve high performance precision manipulation even with very rough models of the manipulation process. With experiments using step and periodic reference signals (in total 204 experiments), we show that the utilization of MPC provides superior performance in terms of accuracy and efficiency comparing to the conventional visual servoing methods.**

I. INTRODUCTION

Dexterous within-hand manipulation has a number of advantages over full-arm manipulation, including working around obstacles and joint singularities, providing additional precision and lower energy/gains due to relatively low inertias of the fingers compared to the full arm, among the others. In conventional methods of within-hand manipulation, the kinematic and dynamic model of the gripper, object and contact models need to be known accurately in order to precisely manipulate the object [1], [2]. Presence of error in any of these models causes performance degradation and often loss of contact between the fingers and the object, and therefore failure of the task.

Methods of utilizing visual feedback for acquiring robustness to model uncertainties have been proposed in the literature for fully-actuated grippers [3]–[7]. The core of these approaches is to use an adaptive visual servoing method: Extending the formulation in [2] to the vision-based control domain, these methods estimate the relationship between actuator velocities and target object velocity (i.e. visual-motor Jacobian) via online observations. Nevertheless, the contact constraints between the fingers and the object are not imposed explicitly in these methods, and undesired, shaky motions in the transient of the estimation due to initial modeling errors may bring a significant risk of dropping the object. Unfortunately, neither the transient characteristics of these algorithms, nor their sensitivity to initial modeling errors are analyzed thoroughly in literature.

Considering precision manipulation with underactuated hands, acquiring a process model is often harder than the fully-actuated cases: the absence of joint measurements in unactuated degrees of freedom and the use of elastic transmission elements (e.g. springs, elastic bands, elastic materials) provide additional challenges to acquiring precise models of these hands. In our previous work, we proposed the first method in literature for vision-based precision manipulation with underactuated hands [8]. The method exploits the contact stability provided by the adaptive underactuation - the elastic elements of the underactuated mechanisms help keep a restoring contact force on the object during the manipulation task. Coupling this property with Precision Manipulation Primitives (a set of actuation rules) and the robustness of the conventional Image-Based Visual Servoing (IBVS) [9] schemes, we have achieved successful within-hand manipulation.

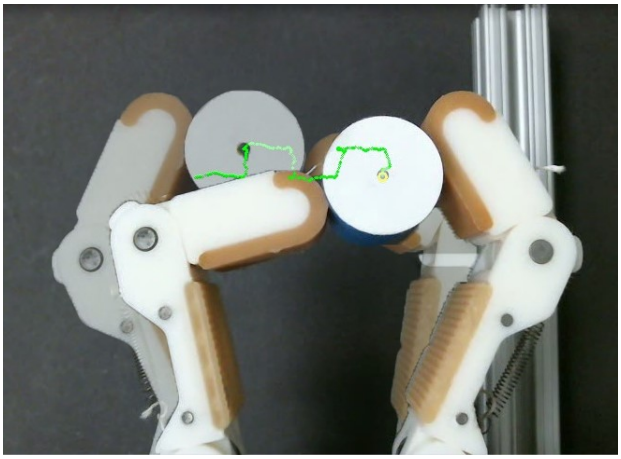

Figure 1. Vision-based within-hand precision manipulation with Model T42 hand.

In this paper, we enhance the performance of this method by formulating the visual servoing problem in a Model Predictive Control (MPC) framework [10]. In robotics, the MPC framework provides effective solutions when there are modeling challenges and when a significant amount of disturbance is acting on the system e.g. mobile robot navigation in dynamic environments [11]–[13], bipedal walking [14], underwater navigation [15], robotic heart surgery [16], [17], control of soft robotic platforms [18] and position/force control schemes [19]. By using MPC for vision-based within-hand manipulation, we introduce additional disturbance rejection, obtain high efficiency, and gain more control over the transient response of the system. All these advantages are demonstrated with experiments using step references and a circular periodic reference signal using objects with various shapes and sizes.

This work was supported by the U.S. National Science Foundation under the grant IIS-1317976.

B. Calli and A. M. Dollar are with the Department of Mechanical Engineering and Material Science, Yale University, CT, USA ({berk.calli;aaron.dollar}@yale.edu).

In the next section, we describe our precision manipulation primitives and their use in a conventional IBVS scheme, which will serve as the baseline for comparing the performance of our MPC-based visual servoing approach. In the third section, we introduce our MPC-based approach for precision manipulation. In Section IV, the experimental methodology and results are presented along with the comparison to the baseline algorithm. Section V concludes the paper.

II. Vision Based Underactuated Precision Manipulation Using Manipulation Primitives

In [8], we derived simple actuation rules, called Precision Manipulation Primitives (PMPs), for the Model T42 hand [20] (Fig. 2). Each PMP is designed to provide orthogonal motion to the target object in Cartesian space. These rules and their use in an IBVS scheme are summarized in this section. The reader can refer to [8] for more detailed explanations of PMPs and their applications.

Assuming the object is already pinch-grasped by the gripper (elastic elements in both fingers are preloaded) and considering only positioning of the target object in x and y-directions in planar manipulation (neglecting the orientation change in this paper for simplicity), we derive two PMPs. PMP-1 moves the target object along the x-direction of the workspace by moving the actuators in the same amount in the opposite directions (Fig. 2b-c). The motion of the object in y-direction is neglected for this PMP:

$$V_{ox} = K_x \dot{q}_1 = -K_x \dot{q}_2, \tag{1}$$

$$V_{oy} = 0, \tag{2}$$

Here V_{ox} and V_{oy} are the velocities of the object in x and y-axis respectively, $\dot{q}_1$ and $\dot{q}_2$ are actuator velocities and K_x is a positive constant.

PMP-2 moves the object along the y-direction, by moving the actuators in the same amount in the same direction (Fig. 2d-e). The motion of the object in x-direction is neglected for this PMP:

$$V_{ox} = 0, \tag{3}$$

$$V_{oy} = K_y \dot{q}_1 = K_y \dot{q}_2, \tag{4}$$

where K_y is a positive constant. Combining these two PMPs, we obtain the following projection between actuator velocities and target object velocity.

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} 1/K_x & 1/K_y \\ -1/K_x & 1/K_y \end{bmatrix} \begin{bmatrix} V_{ox} \\ V_{oy} \end{bmatrix}. \tag{5}$$

If we denote $\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}$, $\boldsymbol{V_o^{hand}} = \begin{bmatrix} V_{ox} \\ V_{oy} \end{bmatrix}$ and the projection matrix as J_{pri}, we obtain:

$$\dot{\boldsymbol{q}} = \boldsymbol{J_{pri}} \boldsymbol{V_o^{hand}}. \tag{6}$$

The projections obtained by PMPs are rough estimates for the kinematic model of the manipulation process. Nevertheless, IBVS algorithms are robust to kinematic model uncertainties. Moreover, taking advantage of the elastic elements of the adaptive hand that maintains contact forces on the object during the process, successful within-hand manipulation can be achieved.

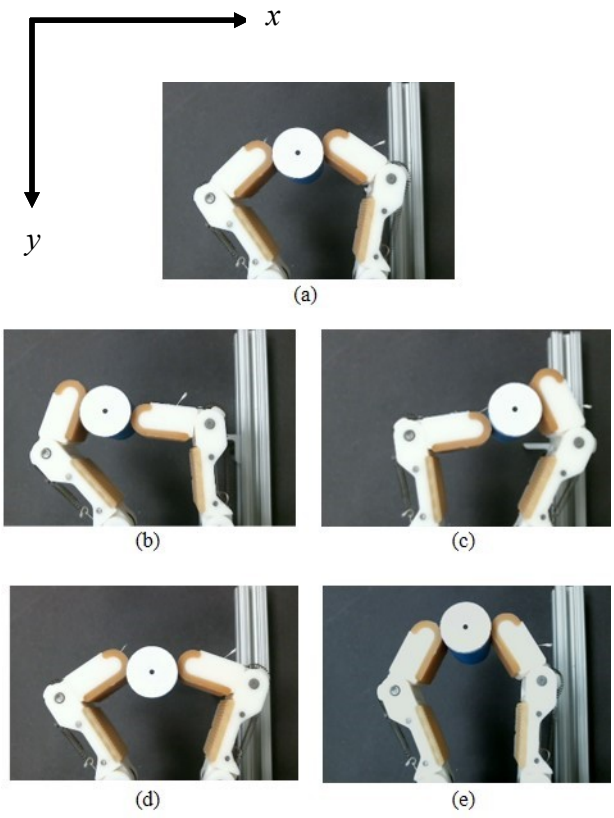

Figure 2. Precision Manipulation Primitives (PMPs) for Model T42 hand. (a) Initial position, (b) and (c) the motion of the object when actuators are moved to the opposite directions in the same amount (PMP-1), (d) and (e) the motion of the object when the actuators are moved in the same direction in the same amount (PMP-2).

The projection in (6) is used in IBVS as follows: The IBVS rule is given by

$$\boldsymbol{V_o^{Cam}} = -\boldsymbol{\lambda} \boldsymbol{J_{int}^{+}} \boldsymbol{e}, \tag{7}$$

where $\boldsymbol{e}$ is the feature error, $\boldsymbol{J_{int}^{+}}$ is the pseudo inverse of the interaction matrix, and $\boldsymbol{\lambda}$ is the diagonal gain matrix. This rule generates velocity references in image space, which can be projected to hand coordinate frame with the Jacobian matrix $\boldsymbol{J_{Cam}^{hand}}$:

$$\boldsymbol{V_o^{hand}} = \boldsymbol{J_{Cam}^{hand}} \boldsymbol{V_o^{Cam}}. \tag{8}$$

The references are then projected to the actuator velocities with

$$\dot{\boldsymbol{q}} = \boldsymbol{J_h} \boldsymbol{J_{contact}^{finger}} \boldsymbol{J_{object}^{contact}} \boldsymbol{V_o^{hand}}. \tag{9}$$

where $\boldsymbol{J_{object}^{contact}}$ is the projection of object velocities to the object-finger contact locations, and $\boldsymbol{J_{contact}^{finger}}$ convert the contact location velocities to finger velocities, both expressed in the hand coordinate frame. $\boldsymbol{J_h}$ is the hand Jacobian that transfers finger velocities to actuator velocities. The $\boldsymbol{J_{pri}}$ matrix in (6) replaces the role of the matrices $\boldsymbol{J_h}$, $\boldsymbol{J_{contact}^{finger}}$ and $\boldsymbol{J_{object}^{contact}}$.

$$\dot{\boldsymbol{q}} = -\lambda \boldsymbol{J}_{\boldsymbol{pri}} \boldsymbol{J}_{\boldsymbol{cam}}^{\boldsymbol{hand}} \boldsymbol{J}_{\boldsymbol{int}}^{+} \boldsymbol{e}. \quad (10)$$

In this paper, instead of using the IBVS rule in (7), we utilize the MPC framework for minimizing the performance inefficiency that can be caused by the simple but inaccurate representation of $\boldsymbol{J}_{\boldsymbol{pri}}$. The next section presents the implementation of an MPC algorithm for vision-based within-hand manipulation and discusses its advantages. Section IV compares the performance of MPC with the IBVS rule in (7) with experiments.

III. Vision Based Model Predictive Control

Typically in Model Predictive Control (MPC) framework, the control signal is obtained by solving an optimization problem considering the future time instances of the process in each control step. The formulation of the problem includes a system model, a disturbance model and the constraints of the task. Using the models, the future states of the system are predicted, and the error between the future reference and states values are minimized also considering the additional constraints and weight parameters. We utilize Generalized Predictive Control formulation in which the cost function that is aimed to be minimized is defined as

$$A(\boldsymbol{U}) = \sum_{j=1}^{N} \delta(j)[\hat{y}(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N} \gamma(j)[\Delta u(t+j-1)]^2. \quad (11)$$

Here $\hat{y}(t+j|t)$ is the predicted state of the system at the future instance $t+j$ calculated at time t (considering both the system model and the effect of the disturbance), $w(t+j)$ is the value of the reference signal at time $t+j$, $\delta(j)$ and $\lambda(j)$ are the weighting parameters, N is the optimization window size and

$$\Delta u(t) = u(t) - u(t-1) \quad (12)$$

$$\boldsymbol{U} = [u(t), u(t+1), \dots u(t+j-1)] \quad (13)$$

where $u(t)$ is the system input. As traditionally used in visual servoing framework, $u(t)$ is a velocity reference to the system.

By minimizing the cost function in (11), the error between the reference and the state is aimed to be decreased to zero, while minimizing the effect of disturbance and penalizing the total control effort. The parameters $\delta(j)$ and $\gamma(j)$ determines the characteristics of the convergence. We chose to use an exponential weight for $\delta(j)$ as

$$\delta(j) = \alpha^{N-j}, \quad (14)$$

where $0 < \alpha < 1$. By this way the later values of the error are penalized more than the earlier values, and a smooth convergence behavior can be achieved. We chose to use a constant positive $\gamma(j)$ value, which penalizes the actuator efforts equally throughout the optimization window.

The state prediction $\hat{y}(t+j|t)$ is calculated with a sum of the system model component y_s and the estimate of the disturbance on the system y_d as

$$\hat{y}(t+j|t) = y_s(t+j|t) + y_d(t+j|t). \quad (15)$$

We use a step response type system model, which is the sum of the effects of the efforts applied to the current state measurement:

$$y_s(t+j|t) = y(t) + \sum_{i=1}^{j} g_i \Delta u(t+i). \quad (16)$$

Here g_i are the model parameters and y(t) is the latest measured state of the system. In our implementation, we use a single g_i value (i.e. $g_1 = g_2 \dots = g_j$) that we obtained experimentally by operating the system at the center of the workspace.

The effect of disturbance (y_d) can be incorporated in various ways. If the effect of disturbance is not directly measurable or predictable for the future values of the system state, then it can be assumed that the effect of disturbance at the last control step will continue to affect in the same way in the future steps. In this case, the disturbance can be estimated as the difference between the last measured state and the effect of control input on the previous state:

$$y_d(t+j|t) = y(t) - (y(t-1) + g_1 \Delta u(t-1)). \quad (17)$$

Alternatively, if the reference signal is periodic, the disturbance on the system is consistent throughout the periodic cycle and the system already completed one full period, then the effect of the disturbance measured at the previous cycle can be used as an estimate

$$y_d(t+j|t) = y(t) - (y(t-\tau) + g_1 \Delta u(t-\tau)), \quad (18)$$

where τ is the number of control cycles that correspond to one period of the reference signal. Such an approach is also used in an application of MPC to robotic heart surgery where the disturbances affect the system periodically [17].

With such a formulation of state prediction, every deviation from y_s will be considered as disturbance and will be minimized by the optimization procedure.

The optimization problem is

$$arg \min_{\boldsymbol{U}} A(\boldsymbol{U}), \quad (19)$$

which has the following explicit solution:

$$\boldsymbol{U} = (\boldsymbol{G}^T \boldsymbol{G} + \gamma \boldsymbol{I})^{-1} \boldsymbol{G}^T (\boldsymbol{W} - \boldsymbol{F}), \quad (20)$$

where $\boldsymbol{I}$ is an NxN identity matrix, and

$$\boldsymbol{G} = \begin{bmatrix} \alpha^{N-1} g_1 & 0 & 0 & \cdots & 0 \\ \alpha^{N-1} g_1 & \alpha^{N-2} g_2 & 0 & \cdots & 0 \\ \alpha^{N-1} g_1 & \alpha^{N-2} g_2 & \alpha^{N-3} g_3 & & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ \alpha^{N-1} g_1 & \alpha^{N-2} g_2 & \alpha^{N-3} g_3 & \cdots & g_N \end{bmatrix}, \quad (21)$$

$$\boldsymbol{F} = \begin{bmatrix} y_d(t+1|t) + y(t) \\ y_d(t+2|t) + y(t) \\ y_d(t+3|t) + y(t) \\ \vdots \\ y_d(t+N|t) + y(t) \end{bmatrix}, \quad (22)$$

$$\boldsymbol{W} = \begin{bmatrix} w(t+1) \\ w(t+2) \\ w(t+3) \\ \vdots \\ w(t+N) \end{bmatrix}. \quad (23)$$

Even though control input is calculated for the next N control steps, only the first control input, $u(t)$, is sent to the system, and the optimization problem is solved again with

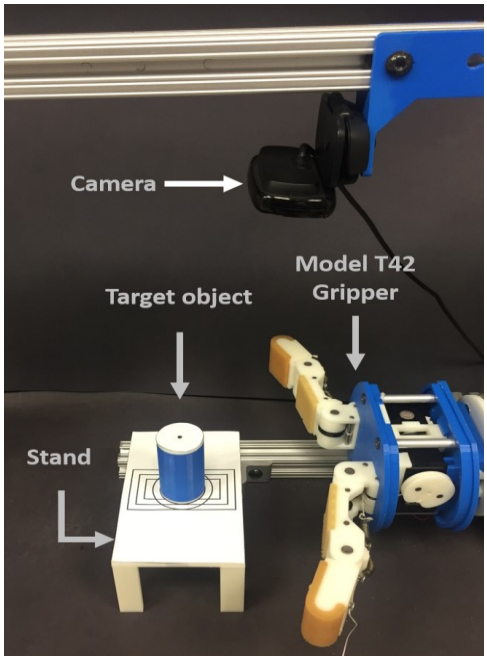

Figure 3. Experimental setup.

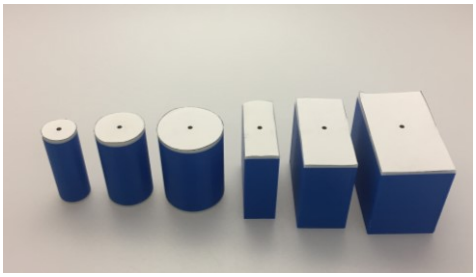

Figure 4. Objects used in the experiments. Cylinders with 2, 3 and 4 cm diameters, and rectangular prisms with dimensions 2x4, 3x5 and 4x6 cm. The objects' weights vary between 12 g and 75 g.

newly acquired data in the next step. $u(t)$ is projected to the actuator space with (6) which is obtained by using the PMPs.

The advantages of using this MPC formulation over conventional IBVS rule are multifold. First, MPC takes into account the effect of disturbances on the system; any deviation from the step response model will be considered as a disturbance and will be minimized. By this, we do not only overcome the friction effects, but also the imperfections of the system model. Second, we have much more control over the transient response of the system than we had with IBVS due to the additional parameters that allows us to achieve smooth transitions. Third, additional constraints can be added to this optimization problem such as workspace constraints and task specific constraints, which could be necessary to impose for many precision manipulation scenarios. However, an explicit solution with additional constraints may not always exist and iterative solvers may be needed. In this case, high computational power may be necessary as this optimization is run in every control step.

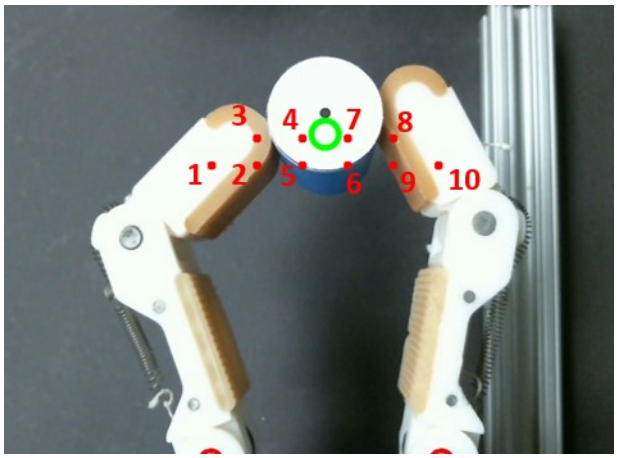

Figure 5. Step references (red) and the circular reference (green) used in the experiments.

IV. Experimental Results

The performance of the vision-based MPC algorithm is tested and compared to the IBVS algorithm and the adaptive method presented in [8] using a Model T42 gripper and various reference signals and target objects. The experimental setup, the target objects and reference signals can be seen in Fig. 3, 4 and 5 respectively. The camera is set to 512x1024 pixels resolution and 30 fps capture rate. The distance of the camera to the object top surface is 18.5 cm. In each experiment, the object is placed on the stand, it is pinch-grasp by the gripper, and then the stand is removed so that the object is manipulated without the plane support. The positons of the target objects are kept consistent in all experiments with a template on the stand, and the initial position of the stand was aligned with a static fixture.

A. Experiments with Step References

A sequence of step references are applied to the system for assessing the performance of our controller. For the cylindrical objects, the object starts at set point 1 presented in Fig. 5, and set points 2-10 are applied sequentially. When the error remains 2 pixels or less for 1 second, the next set point is sent to the system. For the rectangular objects, the same procedure is applied except that the objects start at set point 5, and points 6-8 are applied as the other set points are not realizable due to uncontrolled flip that occurs while servoing from the initial grasp configuration to these points. For each object, IBVS method, adaptive method proposed in [8] and MPC method are applied. For the MPC method, the optimization window size is set to 50. For each method, the experiment is repeated 5 times, and the completion time and total travel distance are presented for cylindrical and rectangular objects in Fig. 6 and Fig. 7 respectively. Also an example of trajectories of the target object for each algorithm can be seen in Fig. 8. From these figures, it can be seen that the MPC method is almost always faster than the conventional IBVS method and the adaptive method in [8] to complete visiting the set points. It also follows a more efficient (shorter) path while travelling between the set points. This is due to the disturbance rejection mechanisms.

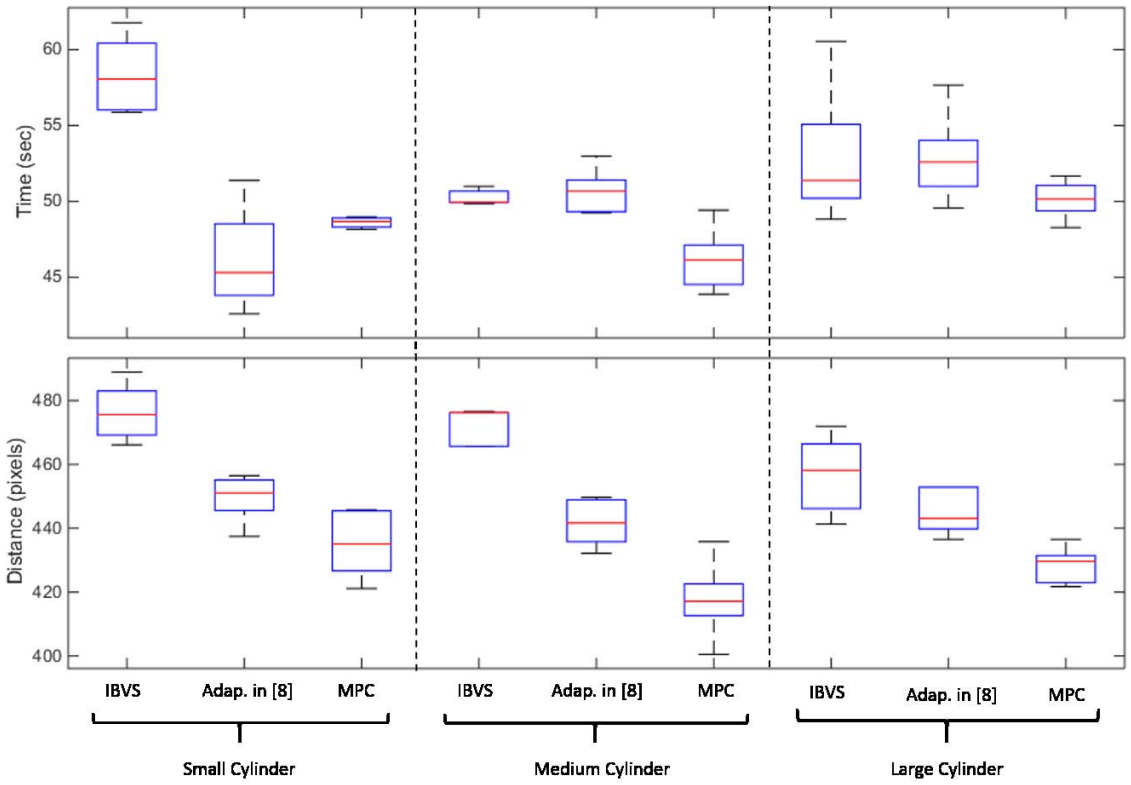

Figure 6. Experiments with the cylindrical objects. Time elapsed to complete the task and total distance traveled are presented. Each box represents five experiments with the same object.

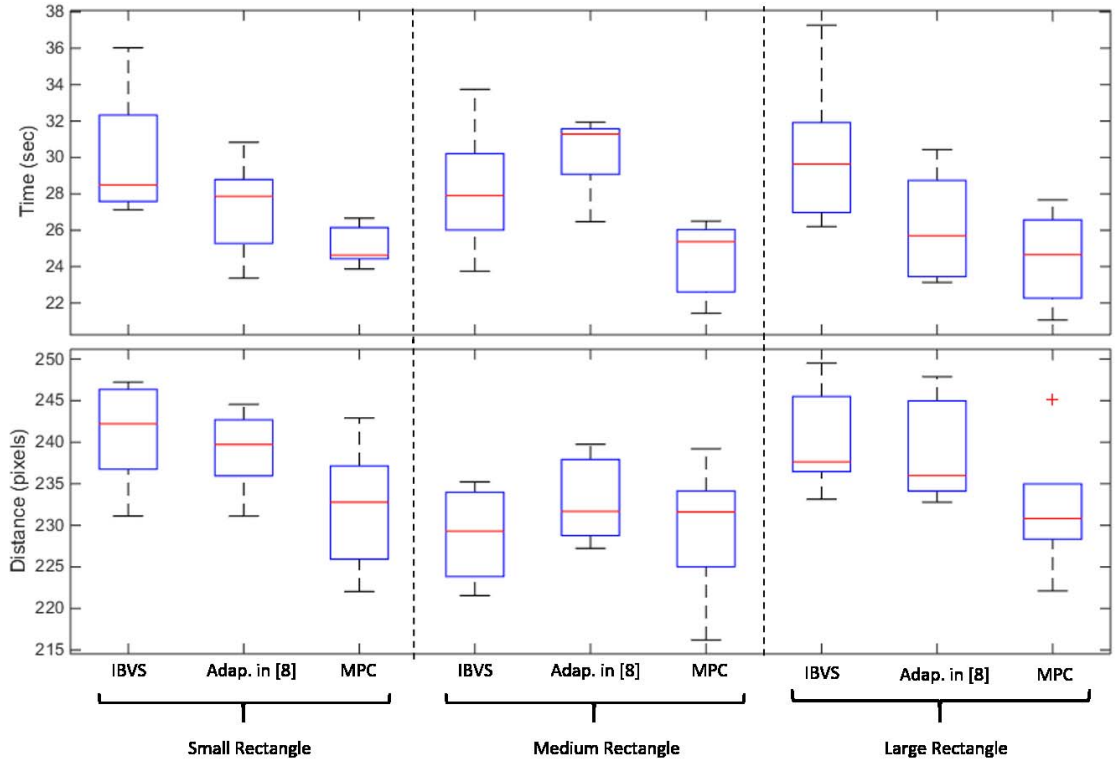

Figure 7. Experiments with the rectangular objects. Time elapsed to complete the task and total distance traveled are presented. Each box represents five experiments with the same object.

Table 1. Experimental results with the circular reference. Average errors are presented in pixels together with improvements with respect to the traditional IBVS method in percentages.

Target object	IBVS	Adapt. in [8]	MPC	MPC periodic
Small Cylinder	2.6	2.7 (-4%)	2.5 (4%)	2.3 (13%)
Medium Cylinder	2.7	2.8 (-4%)	2.4 (13%)	2.2 (22%)
Large Cylinder	2.7	3.1 (-12%)	2.4 (12%)	2.4 (12%)
Small Rectangle	2.8	2.9 (-3%)	2.7 (4%)	1.8 (56%)
Medium Rectangle	3.1	3.1 (0%)	2.7 (15%)	2.6 (19%)
Large Rectangle	2.9	3.0 (-3%)	2.9 (0%)	2.8 (%4)

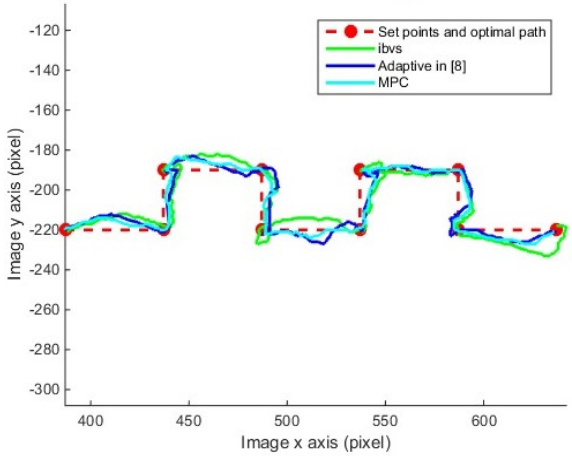

Figure 8. Trajectory followed by the object while set points are supplied sequentially. Results with IBVS, the adaptive method presented in [8] and MPC algorithm.

B. *Experiments with Circular Reference*

The circular reference shown in Fig. 5 with 15 pixel radius and period of 20 secs is applied to the system for three periods. For this set of experiments, MPC with two different disturbance estimation are evaluated: In the first case, the effect of the disturbance are calculated by the information of the previous control step as in (17) (the same estimation used in Section IV.A). In the second case, the effect of disturbance is calculated by the information of the previous period of the periodic motion as in (18).

The performance of the algorithms are evaluated with all the target objects and average error values over the three periods are presented in Table 1. Here, it is observed that the adaptation method in [8] degrades the performance. The reason is that this method necessitates large amount of motions for estimating the velocity direction and for correcting it; small motions in the image space results with quantization errors, and correction cannot be conducted accurately. MPC methods, however, improve the performance of the system significantly. Here we observe that the periodic MPC enhances the tracking performance even better as it has a more accurate estimation of the disturbance for the future steps of the optimization.

V. CONCLUSION

In this paper we proposed the use of a MPC method with vision feedback in order to achieve high performance within-hand precision manipulation. Comparing to the traditional IBVS method, MPC provides more control in the transient response of the system by the parameters introduced in the optimization problem. The experimental results show that MPC provides faster and smoother step response and more precise signal tracking performance. If the reference signal is periodic, even better disturbance rejection can be achieved which increases the precision.

As a future work, we aim to introduce workspace and task specific constraints to the objective function of the MPC in order to realize efficient dexterous manipulation applications.

REFERENCES

[1] A. Bicchi, C. Melchiorri, and D. Balluchi, "On the mobility and manipulability of general multiple limb robots," *IEEE Trans. Robot. Autom.*, vol. 11, no. 2, pp. 215–228, 1995.

[2] A. M. Okamura, N. Smaby, and M. R. Cutkosky, "An overview of dexterous manipulation," in *Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA)*, 2000, vol. 1, pp. 255–262.

[3] C. C. Cheah, H. Y. Han, S. Kawamura, and S. Arimoto, "Grasping and position control for multi-fingered robot hands with uncertain Jacobian matrices," in *Proceedings of the 1998 IEEE International Conference on Robotics & Automation*, 1998, pp. 2403–2408.

[4] M. Jägersand, O. Fuentes, and R. Nelson, "Acquiring visual-motor models for precision manipulation with robot hands," in *Computer Vision (ECCV'96)*, 1996, no. 1, pp. 603–612.

[5] M. Jagersand, O. Fuentes, and R. Nelson, "Experimental evaluation of uncalibrated visual servoing for precision manipulation," in *Proceedings of the 1997 International Conference on Robotics and Automation (ICRA)*, 1997, vol. 4, pp. 2874–2880.

[6] Y. Zhao and C. C. Cheah, "Neural network control of multifingered robot hands using visual feedback.," *IEEE Trans. Neural Networks*, vol. 20, no. 5, pp. 758–67, 2009.

[7] L. A. Munoz, "Robust dexterous manipulation: a methodology using visual servoing," *Proc. 1998 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 1, no. October, pp. 292–297, 1998.

[8] B. Calli and A. M. Dollar, "Vision-based precision manipulation with underactuated hands: Simple and Effective Solutions for Dexterity," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1012–1018.

[9] F. Chaumette and S. Hutchinson, "Visual servo control. I. Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, 2006.

[10] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer Science & Business Media, 2007.

[11] G. Franz and W. Lucia, "A Model Predictive Control scheme for mobile robotic vehicles in dynamic environments," in *Proceedings of the 2013 IEEE Conference on Decision and Control*, 2013, pp. 5752–5757.

[12] Z. Li, S. Member, C. Yang, C. Su, and S. Member, "Vision-based Model Predictive Control for steering of a nonholonomic mobile robot," *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 2, pp. 553–564, 2016.

[13] C. J. Ostafew, A. P. Schoellig, T. D. Barfoot, and J. Collier, "Learning-based nonlinear Model Predictive Control to improve vision-based mobile robot path tracking," *J. F. Robot.*, vol. 33, no. 1,

pp. 133–152, 2015.
[14] M. J. Powell, E. A. Cousineau, and A. D. Ames, “Model Predictive Control of underactuated bipedal robotic walking,” in *Proceedings of the 2015 IEEE International Conference on Robotics and Automation*, 2015, pp. 5121–5126.
[15] S. Heshmati-Alamdari, A. Eqtami, G. C. Karras, D. V. Dimarogonas, and K. J. Kyriakopoulos, “A self-triggered visual servoing model predictive control scheme for under-actuated underwater robotic vehicles,” in *Proceedings of the 2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 3826–3831.
[16] R. Ginhoux, J. A. Gangloff, M. F. de Mathelin, L. Soler, M. M. A. Sanchez, and J. Marescaux, “Beating heart tracking in robotic surgery using 500 Hz visual servoing, Model Predictive Control and an adaptive observer,” in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA)*, 2004, vol. 1, pp. 274–279.
[17] R. Ginhoux, J. Gangloff, M. de Mathelin, L. Soler, M. M. Arenas Sanchez, and J. Marescaux, “Active filtering of physiological motion in robotized surgery using predictive control,” *IEEE Trans. Robot.*, 2005.
[18] C. M. Best, M. T. Gillespie, P. Hyatt, L. Rupert, V. Sherrod, and M. D. Killpack, “A new soft robot control method using Model Predictive Control for a pneumatically actuated humanoid,” *IEEE Robot. Autom. Mag.*, vol. 23, no. August, pp. 75–84, 2016.
[19] J. de la C. Cárdenas, A. S. García, S. S. Martínez, J. G. García, and J. G. Ortega, “Model predictive position/force control of an anthropomorphic robotic arm,” in *Proceedings of the 2015 IEEE International Conference on Industrial Technology (ICIT)*, 2015, pp. 326–331.
[20] L. U. Odhner, R. R. Ma, and A. M. Dollar, “Open-loop precision grasping with underactuated hands inspired by a human manipulation strategy,” *IEEE Trans. Autom. Sci. Eng.*, vol. 10, no. 3, pp. 625–633, 2013.