Optimal Grasping using Visual and Tactile Feedback

Akio NAMIKI Masatoshi ISHIKAWA

Department of Mathematical Engineering and Information Physics
University of Tokyo
Tokyo 113, Japan
namik@k2.t.u-tokyo.ac.jp

Abstract

Sensor feedback and sensor fusion are indispensable for working in the real world. In this paper a robot hand grasping method is proposed. This method uses visual and tactile sensor information. First, grasping evaluation functions is proposed, which is derived from the relationships between a hand and an object. Then a control method is proposed which uses visual and tactile feedback. In this method pre-shaping and grasping are executed smoothly and optimally. Experimental results are presented and a "1ms Sensory-Motor System" is introduced as the experimental system for sensor fusion.

Key Words: *grasping, pre-shaping, multisensor fusion, sensor feedback, vision, tactile sensor*

1 Introduction

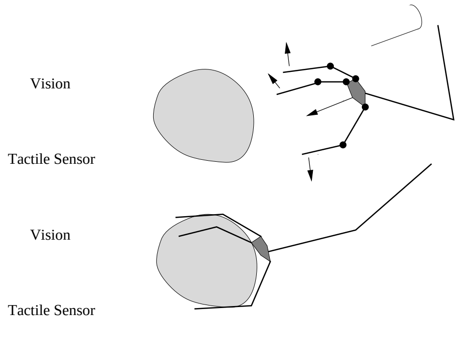
Considering robot hand grasping process in the real world, a manipulated object and environment are mostly unknown. Then there is the possibility that some accidents happen while grasping. A human being can easily solve these problems. For example we can grasp a moving object by bringing our hand near it. Then we can hold an object in darkness using its tactile impression. To execute these excellent grasp faculties a grasping control method using sensor feedback is needed. By using external sensory information, the robot system can cope with many problems.

Most early work to solve grasping problems was based on off-line computing [2, 3]. In these systems grasping is not robust and computing power is very expensive because all information should be observed before grasping. This is not realistic condition in the real world. Then there are some works based on on-line computing. Trinkle gives an estimation in frictionless enveloping grasping [ ]. But this is not a method to search optimal grasping actively. Then Coelho gives a method to search optimal grasping [5]. But sensor feedback is not sufficiently considered in this method. It is necessary to make a grasping method using real-time sensor feedback. Then it is more effective to use many sensor feedbacks and sensor fusion algorithms [1].

We address the problem of a grasping method using sensor feedbacks, particularly visual and tactile feedback. In the next section we discuss the relationship between grasping process and sensor feedbacks.

2 Grasping using Sensor Feedback

The grasping process can be classified into two phases (Figure 1).



$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_{m_c} \end{bmatrix} \in \boldsymbol{R}^{3m_c \times 6},\ \boldsymbol{l} = \begin{bmatrix} l_1 \\ l_2 \\ \vdots \\ l_{m_c} \end{bmatrix} \in \boldsymbol{R}^{3m_c}.$$

Eqn. (6) is the basic equation of grasping. If $l_k = 0$, the contact constraint $\dot{l}_k \geq 0$ is imposed. Satisfying $\dot{l}_k = 0$, the contact point

If the functions $\varepsilon_1^{velocity}$ and $\varepsilon_2^{velocity}$ are minimized, grasping is realized that contact velocity at each contact point become minimum when the object is moved. This means that it is difficult to separate fingers from an object. Then if there are enough contact points, object motion is constrained. This corresponds to the "form closure" without slip. If the functions ε_1^{force} and ε_2^{force} are minimized, grasping is realized that contact forces become minimum when external force is applied to the object. This means that grasp is robust against disturbances. Then if there are enough contact points, object motion is constrained. This correspond to the "force closure" without friction. If the functions $\varepsilon_1^{passive}$ and $\varepsilon_2^{passive}$ are minimized, grasping is realized that passive joint torques are minimized when contact forces are applied to fingers. If the functions ε_1^{active} and ε_2^{active} are maximized, grasping is realized that joint torques applied to the object actively are maximized.

By combining these functions, the functions for desirable grasping is defined as:

Function of Grasp Condition

$$\begin{aligned}\varepsilon^{grasp} = f(&\varepsilon_1^{force} \quad \varepsilon_2^{force}, \varepsilon_1^{passive} \quad \varepsilon_2^{passive}, \\ &\varepsilon_1^{velocity} \quad \varepsilon_2^{velocity}, \varepsilon_1^{active} \quad \varepsilon_2^{active}), \end{aligned} \tag{18}$$

where function f() expresses the linear summation of each element or each inverse element.

4.2 Function of Contact Distance

We evaluate the contact distance on the virtual contact points with the following function:

Function of Contact Distanc

$$\varepsilon^{contact} = \frac{1}{2m_c}\boldsymbol{l}^T M_l \boldsymbol{l}, \tag{19}$$

where $M_l \in \boldsymbol{R}^{m_c \times m_c}$ is a weight matrix. If complete contact is achieved on each virtual contact point, the value of this function equals zero.

5 Grasping Algorithm

In this paper we define the "optimal grasping" as the grasping process, in which the evaluation functions about both the grasp condition ε^{grasp} and the contact distance $\varepsilon^{contact}$ are minimized. In this section a planning algorithm for optimal grasping is proposed. This algorithm uses the evaluation functions in Section . In this algorithm position and shape of hand is gradually changed using sensor feedback.

5.1 Optimal Planning using Evaluations

Assuming that the object does not move during grasp motion, namely $\dot{\boldsymbol{c}} = 0$. Substituting this into Eqn. (6), the equation of virtual contact is derived:

$$G\dot{\boldsymbol{q}} = \dot{\boldsymbol{l}}, \tag{20}$$

where $G = N^T[J\ H] \in \boldsymbol{R}^{m_c \times (m_\theta + 6)}$, $\boldsymbol{q} = (\boldsymbol{\theta}^T, \boldsymbol{p}^T)^T \in \boldsymbol{R}^{m_\theta + 6}$. We call a vector $\boldsymbol{q}$ the "hand parameter", which is a controllable parameter.

By using the generalized inverse of G, the equation of the hand parameter is derived:

$$\dot{\boldsymbol{q}} = G^+\dot{\boldsymbol{l}} + (I - G^+G)\dot{\boldsymbol{x}}, \tag{21}$$

where the first term is the particular solution and depends on the derivative of the contact distance. The second term is the solution of $G\dot{\boldsymbol{q}} = 0$ and $\dot{\boldsymbol{x}}$ is an arbitrary vector. Each contact distance is invariable during motion by the second term.

We adopt Eqn. (21) for planning grasp. By using evaluation functions ε^{grasp} and $\varepsilon^{contact}$ in Section , the planning in each step is calculated as

$$\begin{aligned}\boldsymbol{q}^{k+1} &= \boldsymbol{q}^k + \beta^k \boldsymbol{\lambda}^k, \\ \boldsymbol{\lambda}^k &= -\gamma^c G^+ \frac{\partial \varepsilon}{}\end{aligned} \tag{22}$$

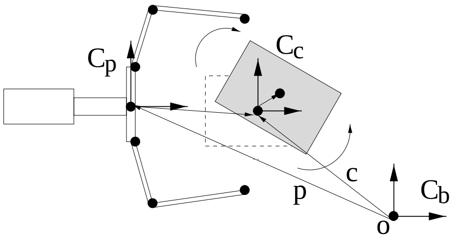


Figure 5: Compensation for Object Motion

The constraint of compensation is that the matrix T_p^c is constant. Under this constraint the derivative of Eqn. (2) is shown as

$$\dot{T}_p^b = \dot{T}_c^b T_p^c. \tag{25}$$

The homogeneous matrices $\dot{T}_p^b$ and $\dot{T}_c^b$ can be respectively expressed by the palm wrench vector $\boldsymbol{p}$ and the wrench vector of the center of gravity $\boldsymbol{c}$. This relationship is shown as

$$\dot{\boldsymbol{p}} = V_c^p \dot{\boldsymbol{c}}. \tag{26}$$

We add this equation to Eqn. (23) to compensate for object motion as

$$\boldsymbol{q}^{k+1} = \boldsymbol{q}^k + \beta^k \boldsymbol{\lambda}^k, \tag{27}$$

$$\boldsymbol{\lambda}^k = -\gamma^c G^+ M_l \boldsymbol{l} - \gamma^g (I - G^+ G) \frac{\partial \varepsilon}{\partial \boldsymbol{q}}^{grasp} + V_c^q \boldsymbol{\Delta c}, \tag{28}$$

where $V_c^q = \left[O_{m_\theta} \ V_c^{pT}\right]^T$.

5.3 Grasping of Virtual Object Model

If a robot hand is far from a manipulated object, we cannot use the proposed method because virtual contact points do not exist. This problem is solved by considering the “virtual object model” (Figure 6). In this method, an object model is virtually considered near a hand and grasping motion is executed to this model. By bringing the virtual model to the real object while grasping the virtual model, grasping motion and approaching motion can be simultaneously executed.

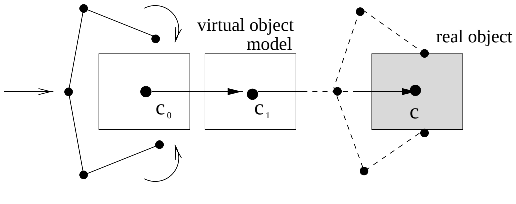


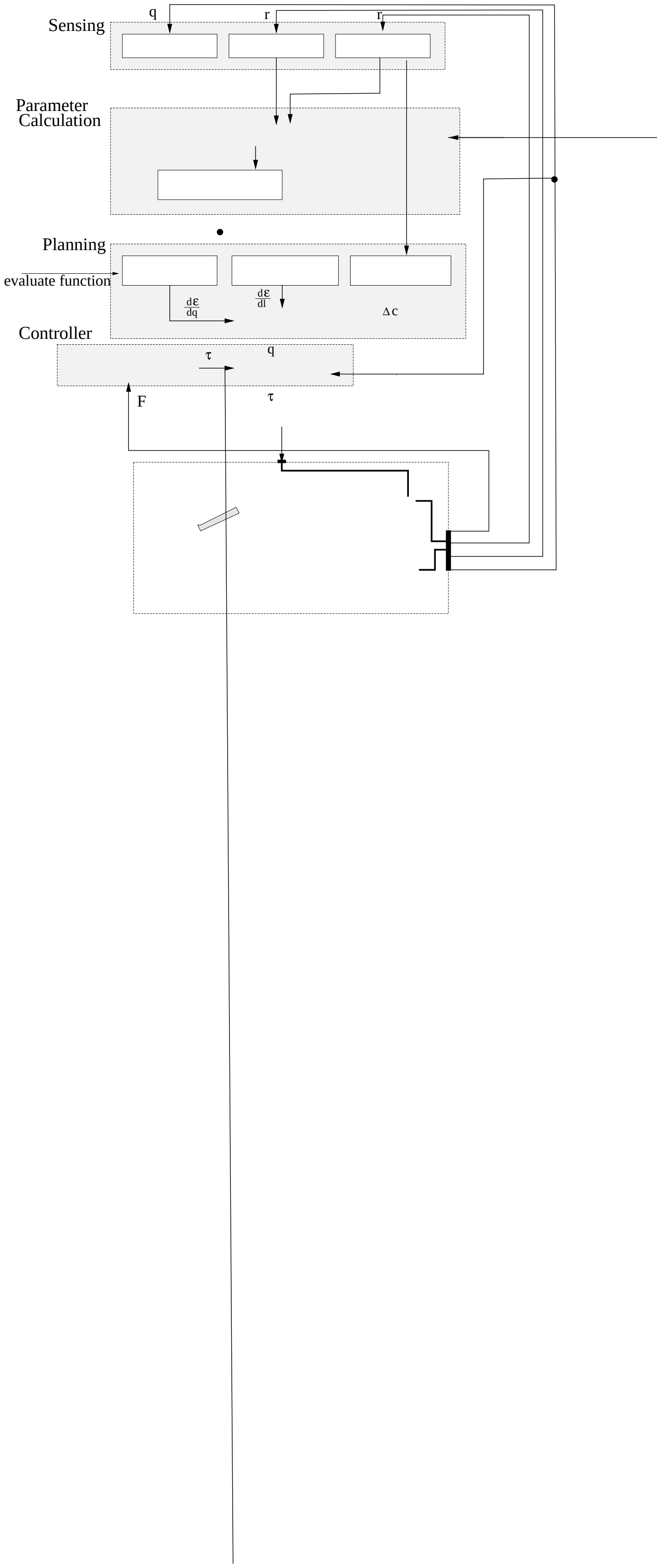
Figure 6: Grasping of Virtual Object Model

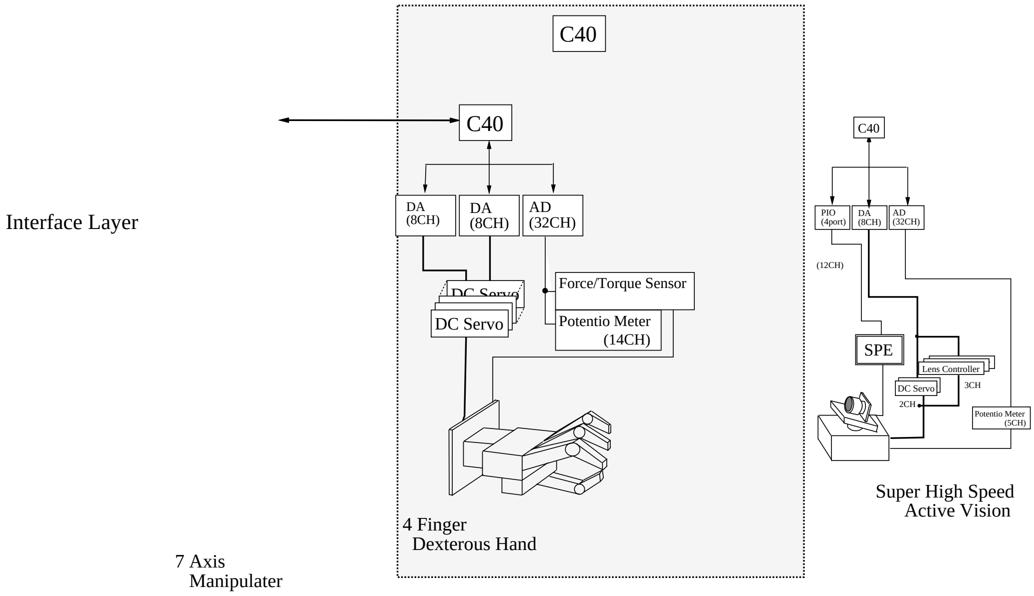
First the orbit and the velocity of the gravity center $\boldsymbol{c}$ of the virtual object model are determined. Then the increment $\boldsymbol{\Delta c}$ should be substituted in Eqn. (27) instead of the increment the motion of the real object. If the real object is moving, the summation is substituted as

$$\boldsymbol{\Delta c} = \boldsymbol{\Delta c}^{real} + \boldsymbol{\Delta c}^{virtual}, \tag{29}$$

where $\boldsymbol{\Delta c}^{real}$ is the increment by the motion of the real object, $\boldsymbol{\Delta c}^{virtual}$ is the increment by the motion of the virtual object.

Grasp motion from a remote place is executed by using this method. It is necessary to consider occlusion between fingers and the object, but this problem will be considered in future.




C40
C40
DA
(8CH)
DA
(8CH)
AD
(32CH)
DC Servo
DC Servo
Force/Torque Sensor
Potentio Meter
(14CH)
4 Finger
Dexterous Hand
Interface Layer
7 Axis
Manipulater
C40
PIO
(4port)
DA
(8CH)
AD
(32CH)
(12CH)
SPE
Lens Controller
3CH
DC Servo
2CH
Potentio Meter
(5CH)
Super High Speed
Active Vision

This system consists of a controller and robot systems. In the controller, we adopt a DSP (TMS320C 0) as the processor, which has the ability to execute high speed calculation and to communicate with other processors. We construct a high speed controller with a parallel architecture using a network of DSPs. This network is connected with many interface ports. Through these ports the controller is connected with the robot systems,, which consists of an active-vision system and a hand-arm system.

The active vision system has the high speed visual processing system SPE (Sensory Processing Elements). The experiment of high speed target tracking is executed on this system [10]. The hand-arm system consists of a 7-axes manipulator and a dextrous hand.

Our grasping algorithm needs much computational power and high speed feedback, then our 1ms feedback system is suitable for this experiment.

7 Experiment

7.1 Experimental System

We use the -fingered dextrous hand in the 1ms feedback system. The palm of the hand is fixed and only the fingers can move. The hand has a potentio meter and a force sensor at each joint. By using these sensors the position and the contact force at each finger tip can be measured. Because this hand does not have a tactile sensor now, we use these sensors for obtaining the positions of contact points instead. Assuming only fingertip contact, these parameters can be calculated. For vision we use a CCD camera because high resolution is needed in this experiment. The system is shown in Figure 9.

In this experiment we use the evaluation function of contact distance and the evaluation function of contact velocity. The planning method described in Section 5 is used. Because it is difficult to execute both control and estimation at the same time we adopt the algorithm shown in Figure 10. In this algorithm the time interval of position feedback control and force feedback control is within 3ms. But visual feedback is executed only 1 time evofy1700(steps17000(tar17000(edb1160008(u)-210002J60i5000(p(time)-20000ce)-9.5t1eeded)]TJo-2106tsed)]TJo-2106tsed)ho-2

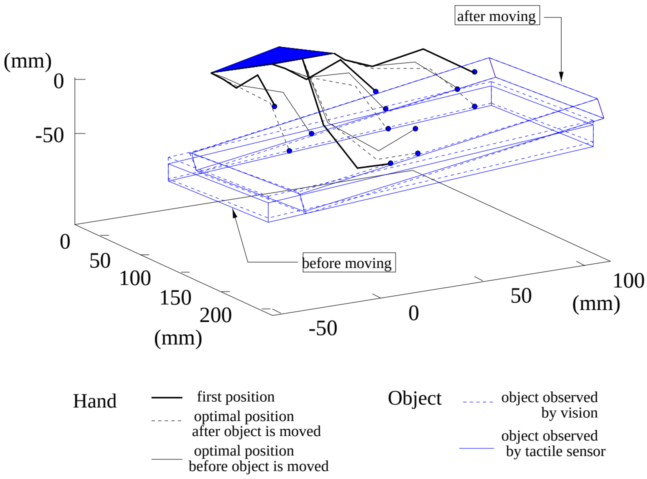


Figure 11: Results of Hand Planning

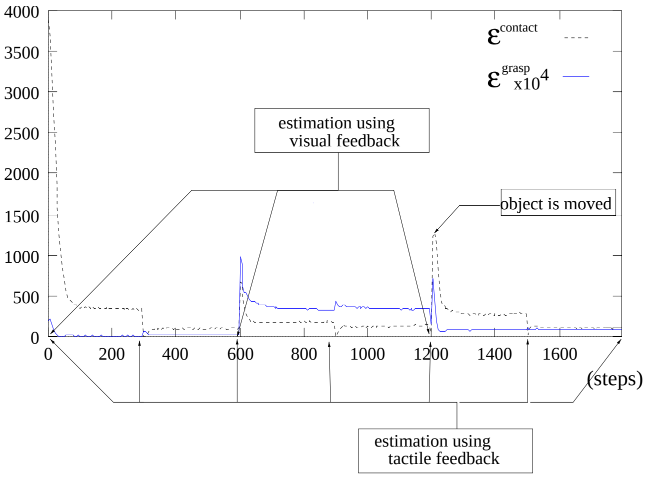


Figure 12: Result of Evaluation Values

constrained against the orientation of the normal vectors of contact planes. This result matches the property of the evaluation function of contact velocity.

Figure 12 shows how evaluation functions vary with time. In this figure both evaluation functions go to zero and these two values are minimized in parallel. When the object is moved, the values of evaluation functions become temporarily high. But new parameters are estimated by visual feedback so that evaluation functions go to zero again. After estimation using visual and tactile information the values of evaluation functions also become high. This is because of estimated errors. However in this case optimization is also executed and these values go to zero again.

These results shows that this system is robust against errors and disturbances because they are compensated by sensor feedback. There is a problem that motion of the hand wings if estimated values swing. To solve this problem it is necessary to use estimation that it can be executed in real-time and whose estimated errors are small. This problem will be considered in future.

In this experiment we cannot show the effectiveness of algorithm in high speed grasping because the sensors are slow. But now a high speed vision chip is being developed in our laboratory. If such high-speed sensors are realized the validity of our algorithm can be demonstrated.

8 Conclusion

A grasping algorithm using visual and tactile feedback is introduced. This algorithm is robust against disturbances and effective in all grasping phases, that is both contact and non-contact phase.

The algorithm consists of two parts. One is an evaluation function of grasping. This can be be easily included in the sensor feedback loop and evaluate grasping in both the contact phase and the non-contact phase.

The second part is a planning method of grasping. We propose a new method deriv