# High-Speed, Small-Deformation Catching of Soft Objects Based on Active Vision and Proximity Sensing

Keisuke Koyama, Kenichi Murakami, Taku Senoo, Makoto Shimojo, and Masatoshi Ishikawa

***Abstract*—In this letter, we propose a combination of sensing and control modules for catching soft objects (i.e., a marshmallow and paper balloon) at a high speed with small deformation. A paper balloon and marshmallow are deformed by a small force and they have individual differences in terms of softness. Therefore, model-based, small-deformation catching is difficult. To realize small-deformation catching, we used high-speed sensor-based (vision and proximity) control modules without a deformation model of the soft objects. A high-speed vision-based controller adjusts the positions of the fingertips of a robot to some extent. As the distance to the object decreased, the fingertip positions were accurately controlled by the high-speed, high-precision proximity-based control. Furthermore, the fingertips were stopped by a proximity-based contact detection before the object surface was deformed. Virtual damping control was effective in catching an object whose surface can be easily deformed by a small impact force or vibration of the fingertips, such as a paper balloon.**

***Index Terms*—Perception for Grasping and Manipulation, Sensor-based Control, Multifingered Hands, Proximity sensor.**

## I. Introduction

DYNAMIC throwing and hitting of a rigid ball using a bat [1], [2] have been realized using a high-speed robot hand and arm with 1 ms visual feedback so far. We propose proximity sensor-based contact detection in which "zero distance" is defined as "contact", and we demonstrated the catching of a soft object at a high speed [3]. A combination of vision and proximity feedbacks enables continuous sensing from noncontact to contact states with 1 ms and high resolution without visual occlusion. Therefore, the control algorithm of the robot does not need any estimation process and becomes simple and robust. In this study, we demonstrate the catching of soft objects such as a marshmallow and paper balloon at a high speed and with small deformation, using only simple sensor-feedback control modules.

Small-deformation grasping and manipulation of soft objects are necessary skills in robotic food manufacturing, home-service robots, and agricultural robots. For fast grasping without deformation, the following are necessary: 1. high-resolution, high-speed sensing from a noncontact state to a contact state, 2. decreasing of the impact force and fingertip vibrations at the time of contact, and 3. adaptation to variables such as differences in the softness and size of different objects, and 4. in-hand manipulation with multiple fingers. A robot hand system with the above four abilities has not been developed so far. High-speed and small-deformation grasping and manipulation are very difficult for typical grasping systems such as vision- and tactile-based systems.

In a vision- and tactile-based grasping system, an error in position estimation by the vision sensor becomes the first problem. The second problem is the sensitivity of contact detection of a tactile sensor. If the estimated position is far from the object surface, the fingertip will not touch it and the robot hand will drop the object. On the other hand, if the estimated position is inside the object, the fingertip penetrates the object, and the extent of deformation of the object depends on the sensitivity of contact detection of the tactile sensor. Although high-sensitivity tactile sensors have been proposed [4]–[6], these are currently being researched and developed. Most research focuses on developing sensors and investigating their characteristics. No one has proposed a high-speed grasping strategy with small deformation that exploits high-sensitivity contact detection.

Therefore, in this study, we propose control modules for small-deformation catching of soft objects at a high speed. Overviews of the modules are shown in Fig. 1. We used the high-speed and high-precision proximity sensor [3] in addition to the high-speed multifingered hand and high-speed vision sensor [2], to realize the following: 1. high-speed, high-resolution sensing, 2. reduction in the impact force and vibration, and 3. adaptation to variations in the softness and size of objects. We demonstrate catching of a marshmallow and paper balloon that are exceptionally soft and difficult to catch with small deformation.

## II. Difficulties in High-Speed Soft-Object Catching and Related Work

### A. Difficulties in Marshmallow Catching

The shape of a marshmallow is cylindrical, and the softness is different for each piece. This makes it difficult to catch it at high speeds with small deformation.

Manuscript received September 10, 2018; accepted December 19, 2018. Date of publication January 7, 2019; date of current version January 25, 2019. This letter was recommended for publication by Associate Editor Z. Xiong and Editor H. Ding upon evaluation of the reviewers' comments. *(Corresponding author: Keisuke Koyama.)*

The authors are with the Department of Creative Informatics, University of Tokyo, Bunkyo-ku 1138656, Japan (e-mail: Keisuke_Koyama@ipc.i.u-tokyo.ac.jp; Kenichi_Murakami@ipc.i.u-tokyo.ac.jp; Taku_Seno@ipc.i.u-tokyo.ac.jp; Makoto_Shimojo@ipc.i.u-tokyo.ac.jp; Masatoshi_Ishikawa@ipc.i.u-tokyo.ac.jp).

Digital Object Identifier 10.1109/LRA.2019.2891091

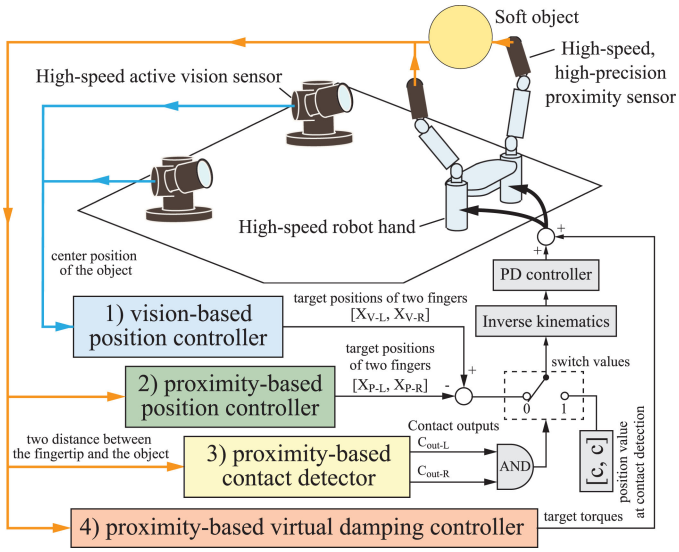


Fig. 1. Control modules for high-speed and small-deformation catching. It consists of 1) a vision-based position controller, 2) proximity-based position controller, 3) proximity-based contact detector, and 4) proximity-based virtual damping controller. Control modules 1) and 2) adjust the two finger positions. The detector detects contact with the object at zero force. Further, damping controller 4) reduces the impact force and vibration of the fingers at the time of contact.

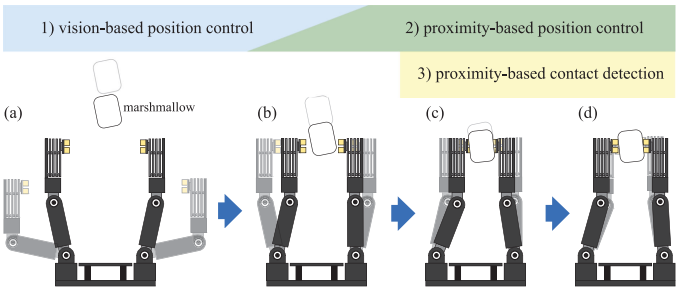


Fig. 2. Marshmallow catching process. For marshmallow catching, we used three control modules: 1) a vision-based position controller, 2) proximity-based position controller, and 3) proximity-based contact detector.

A fresh, soft marshmallow is deformed by 0.8 mm when a 20 g force is applied. On the other hand, a stale and firmer one is deformed by 0.022 mm with the same force [7]. The amount of deformation changes by more than ten times depending on the freshness, and it is not easy to define a useful deformation model for marshmallows.

Furthermore, the contact area depends on the posture of the marshmallow. In case of a small contact area, the output from a tactile sensor during contact will be small. Contact detection before deformation is difficult for the tactile sensor.

To realize catching of the marshmallow, we used three control modules, namely 1) a vision-based position controller, 2) proximity-based position controller, and 3) proximity-based contact detector, as shown in Fig. 2. These control modules do not need a deformation model; they only use the approximate width of the marshmallow. 1) The vision-based position controller adjusts the fingertip position roughly based on the approximate width, and 2) the proximity-based position controller controls the fingertip positions precisely when the object comes close, within 20 mm, to the hand. Further, contact with

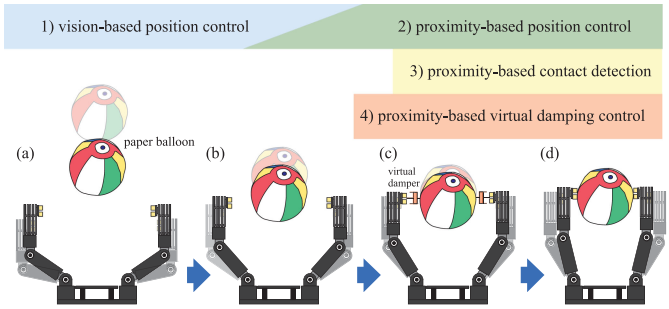


Fig. 3. Paper-balloon catching process. For paper-balloon catching, we used 4) a virtual damping controller in addition to the three control modules.

the marshmallow is detected with zero force by the 3) proximity-based contact detector, and the fingertips are stopped.

### *B. Difficulties in Catching Paper Balloons*

The paper balloon is one of the most easily deformable objects. The deformation is mostly plastic. It is difficult to investigate the relationship between the force and deformation, and defining a deformation model for the paper balloon is not easy. Moreover, a tactile sensor cannot detect contact before deformation because the surface deforms significantly on applying an extremely small force. Even if a fingertip is immediately stopped at the surface, the surface will be deformed considerably owing to the impact force and vibrations of the fingertip.

To catch the paper balloon gently, we used 4) a proximity-based virtual damping controller in addition to the above three control modules, as shown in Fig 3. The proximity-based virtual damping controller decreases the fingertip speed continuously based on the differential value of the detected distance. The impact force and vibration of the fingertips were reduced by the controller and the amount of deformation became small.

### *C. Related Work*

In high-speed catching of rigid objects, many studies used a vision sensor with a detection or estimation algorithm for appropriate catching. Namiki *et al.* [1] developed a high-speed multifinger robot hand and demonstrated the dynamic catching of a rigid object by using the hand and 1 ms visual feedback. They also realized robotic ball juggling with high-speed visual and tactile feedback [8]. Kober *et al.* [9] also demonstrated ball juggling with a humanoid. Bäuml *et al.* [10] presented a robotic ball-catching system having a hand and arm with multiple degrees of freedom. Further, they realized catching with a robot having the most degrees of freedom (two arm: $2 \times 7$DOF, a torso with head: 5DOF, two hands: $2 \times 12$DOF and an omnidirectional base) [11]. Kim *et al.* [12] realized catching in-flight objects with uneven shapes using a programming-by-demonstration approach. They demonstrated catching of various objects such as a hammer and tennis racket through learning from throwing examples and dynamic models of the object and arm motion. Cigliano *et al.* [13] presented robotic ball catching with only an eye-in-hand system (single camera). Salehian *et al.* [14] presented soft catching of a flying object with appropriate follow motion to the object. The motion is expressed as

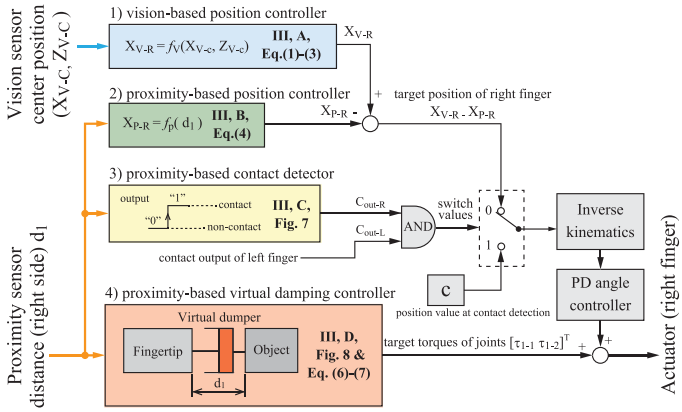


Fig. 4. Block diagram of the control modules of the right finger. We also used the same control modules for controlling the left finger.

a linear parameter varying (LPV) system. They also proposed estimation and approximation methods for the parameters using Gaussian mixture models. Bhole *et al.* [15] proposed a variable stiffness actuator for catching an object that rolls down a ramp without bouncing back.

For grasping soft objects, Cretu *et al.* [16] proposed a segmentation and monitoring method for shape deformations of soft objects grasped by a multifingered robot hand. The method runs fast, but high-speed grasping with small deformation is not realized. Takahashi *et al.* [17] realized grasping of unknown objects (different mass and size of a tofu piece) with robust force and position control based on tactile sensing. Nishimura *et al.* [18] examined the compression behavior of a tofu piece and realized its grasping without breaking by using a fluid-filled fingertip pressure sensor. Krahn *et al.* [19] designed a soft-touch gripper with a variable volume chamber sealed by a thin flexible latex membrane. The gripper can grasp delicate objects such as fruits or vegetables owing to the friction between the membrane and target object. Yamaguchi *et al.* [20] demonstrated several manipulation strategies with vision-based tactile sensing (Finger Vision). They reported that Finger Vision could detect proximity vision and estimate the force required for a marshmallow and paper bird.

Thus, high-speed catching of rigid objects or grasping of soft objects has already been realized by many researchers. However, high-speed catching of soft objects with small deformation has not been achieved. Further, the grasping strategy, sensing, and controls for the catching process have not been proposed yet. This research is the first attempt to realize catching without deformation and propose a control strategy, combining high-speed vision and high-speed, high-precision proximity sensors.

## III. Control Modules and Task Setting

Fig. 4 shows the details of the control block of the right finger. It consists of the following four control modules: 1) a vision-based position controller, 2) proximity-based position controller, 3) proximity-based contact detector, and 4) proximity-based virtual damping controller. Modules 1) and 2) calculate the target position of the fingertip according to the falling position and distance to the object. When contact with

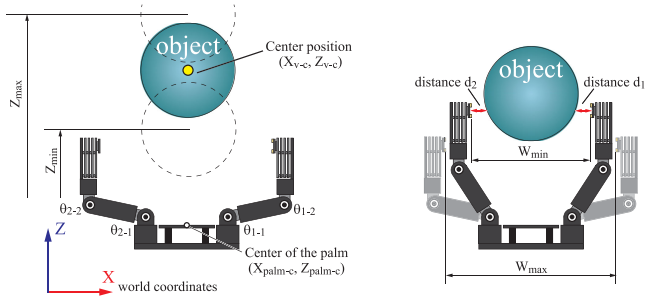


Fig. 5. Task setting for catching. The object moved in the $X-Z$ plane, and the maximum and minimum heights of the object are respectively $Z_{\max}$ and $Z_{\min}$. The center position of the object is $(X_{\mathrm{v-c}}, Z_{\mathrm{v-c}})$, and the center position of the palm is $(X_{\mathrm{palm-c}}, Z_{\mathrm{palm-c}})$. The maximum and minimum opening widths of the fingers are $W_{\max}$ and $W_{\min}$.

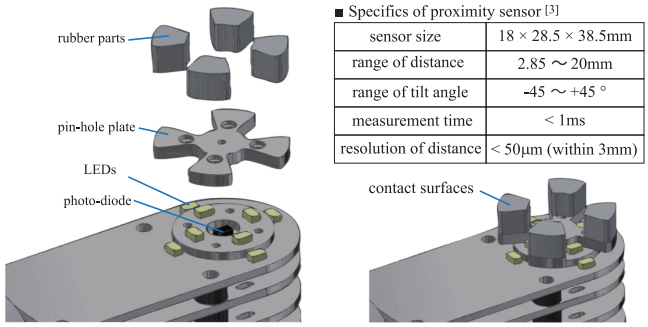


Fig. 6. Overview and specifics of the high-speed, high-precision proximity sensor. Rubber with a thickness of 3 mm is placed on the sensor surface. The sensor can sense objects until contact with the object happens.

the object is detected by module 3), the target position value at the time of contact detection is maintained and the fingertip motion is stopped. Module 4) is continuously operated even after contact to decrease the fingertip vibration and impact force at contact.

As shown in Fig. 5, the high-speed robot hand catches a soft object (marshmallow or paper balloon) dropped in the $X-Z$ plane by controlling the four flexion joints of two fingers. The high-speed vision sensor was set with the world coordinate system. High-speed, high-precision proximity sensors were mounted on each of the two fingers.

### High-speed active-vision sensor

Two active-vision sensors tracked the target object and measured its center position in 3D space. One active-vision sensor consists of a camera (IDP-Express R2000, Photron) and two-axis (pan, tilt) rotation stages actuated with high speeds. The camera was installed on the rotation stages. The vision sensors tracked a target object so that the center position of the object became the center of the image captured by the sensor. Further, the center position of the object was calculated from the image and joint angles of the rotation stages. Details of the camera and image processing are described in Section IV. In the experiment, we used the center position of the object $(X_{\mathrm{v-c}}, Z_{\mathrm{v-c}})$ in the $X-Z$ plane (see Fig. 5, left).

### High-speed, high-precision proximity sensor

Fig. 6 shows the overview and specifics of the proximity sensor. This sensor is a fingertip-sized optical sensor that measures

the tilt angles and distance of the object surface. The sensor measures the distance to the object $d_{\rm i}(i = 1, 2)$ in the $X$ direction (see Fig. 5, right). The detection range includes distances of $2.85 \sim 20$ mm, tilt angles of $-45 \sim +45°$, and measurement times less than 1 ms. Moreover, within 2.85∼3 mm, the resolution is 50 $\mu$m and the peak-to-peak repeat measurement error is 13 $\mu$m [3]. The sensor can detect an object until it contacts the rubber on the sensor surface. In our previous work [3], we proposed contact detection that did not depend on the contact force with only proximity sensing. The first objective of this study is to propose a combination of high-speed vision- and proximity-based controls. Second, we intend to propose simple control modules for catching.

### A. Vision-Based Position Controller

This controller calculates the target positions of two fingers in the hand coordinates from the center position $(X_{\rm v-c}, Z_{\rm v-c})$ of the object. The equations are as follows.

$$X_{\rm V-R} = \frac{W_{\max}}{2} - C_{\rm v1}\frac{(W_{\max} - W_{\min})}{2} - (X_{\rm palm-c} - X_{\rm v-c}) \tag{1}$$

$$X_{\rm V-L} = \frac{W_{\max}}{2} - C_{\rm v1}\frac{(W_{\max} - W_{\min})}{2} + (X_{\rm palm-c} - X_{\rm v-c}) \tag{2}$$

$$C_{\rm v1} = \frac{Z_{\max} - Z_{\rm v-c}}{Z_{\max} - Z_{\min}} \tag{3}$$

Here, the posture of each fingertip is always constant. $X_{\rm V-R}$, $X_{\rm V-L}$ are the controller outputs (position values) of the right finger and left finger in the hand coordinates. $C_{\rm v1}$ is a coefficient of the opening width between the fingers, which changes with the object position in the $Z$ direction. The range of $C_{\rm v1}$ is from 0 to 1.0. $W_{\max}$ is the maximum opening width between two fingers. $W_{\min}$ is the minimum value, and this value is set as the object width (including the error in the size). $X_{\rm palm-c}$ is the center position of the hand palm in terms of world coordinates. With the vision-based position control, the opening width between the two fingers is adjusted according to the drop height and position of the object. In marshmallow catching, only the feedback of the $Z_{\rm v-c}$ was used. On the other hand, in paper-balloon catching, feedback from $X_{\rm v-c}$ and $Z_{\rm v-c}$ was used, because the falling trajectory of the balloon changed slightly owing to air resistance.

The proximity-based controller output, which is described below, was added to the vision-based controller output. The target joint angles were calculated from the total value of these outputs using inverse kinematics.

### B. Proximity-Based Position Controller

When the distance between the finger and object is within 20 mm, the vision-based controller output is corrected based on the distance value determined by the proximity sensor.

$$X_{\rm P-R} = K_{\rm p}\int (d_1 - d_{\rm ref})dt \tag{4}$$

$$X_{\rm P-L} = K_{\rm p}\int (d_2 - d_{\rm ref})dt \tag{5}$$

Here, $X_{\rm P-R}$ and $X_{\rm P-L}$ are the controller outputs (position value) of the fingertip. $K_{\rm p}$ is the feedback gain. We set $K_{\rm p}$ to appropriate values experimentally according to the falling speed of the object. $d_{\rm ref}$ is the target distance equal to the thickness of the rubber on the fingertip (= 3 mm). The error in the distance between the fingertip and object is corrected by the feedback.

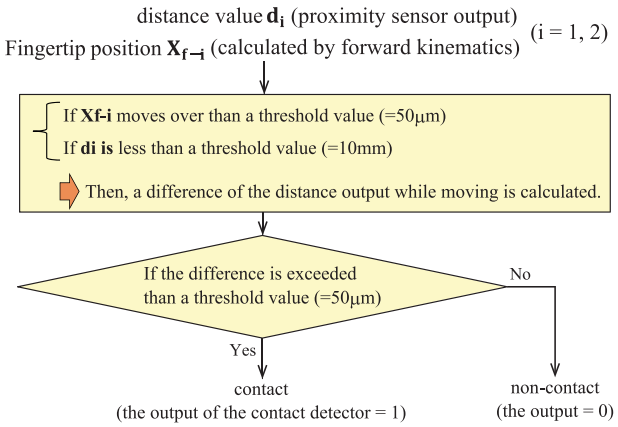


Fig. 7. Flowchart for the contact detector. The detector judges the contact with an object based on the distance value of the proximity sensor and the fingertip position calculated using forward kinematics.

### C. Proximity-Based Contact Detector

We define a distance of zero millimeter as contact. The contact detector outputs "1" (contact) or "0" (noncontact) based on the distance value of the proximity sensor and a change in the fingertip position. Fig. 7 shows the flowchart for the contact detector. In the flowchart, when the fingertip position moves by more than 50 $\mu$m and the distance value is sufficiently small, the difference in the distance output after the fingertip moved is calculated. In the noncontact condition, the difference is almost zero. On the other hand, if the sensor and fingertip make contact, the difference exceeds the threshold value. Therefore, if the difference exceeds the threshold value of 50 $\mu$m, which is the minimum resolution of the proximity sensor, the contact detector judges that contact with the object is made. This condition enables robust contact detection for various soft objects having a smooth or uneven surface.

### D. Proximity-Based Virtual Damping Controller

When the object has damping characteristics and elasticity, the impact force and vibration at contact are low. However, in the case of an extremely soft plastic object such as the paper balloon, the damping characteristic of the object is extremely low. Further, the damping force of the rubber is zero because rubber does not deform. Therefore, the surface of the object

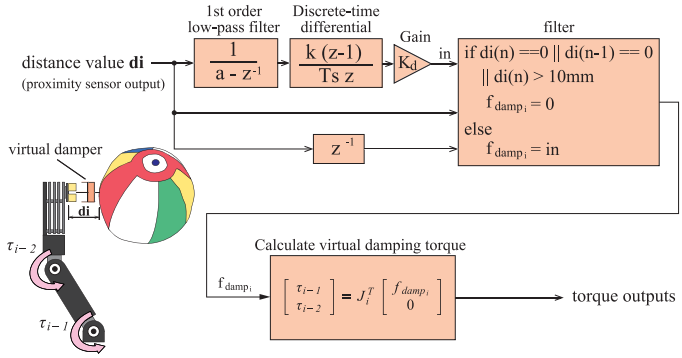


Fig. 8. Block diagram of the proximity-based virtual damping control.

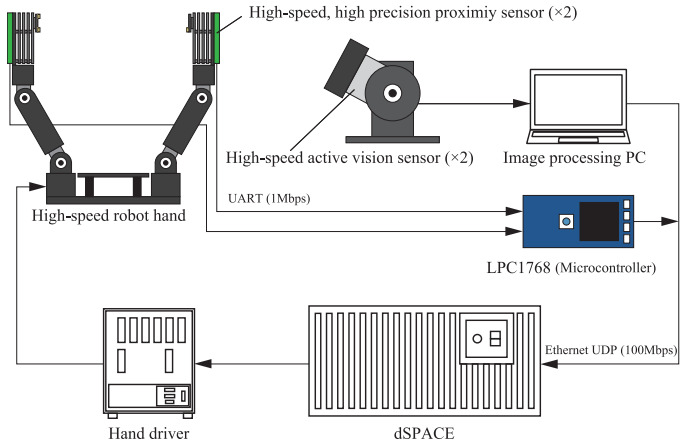


Fig. 9. High-speed hand system with high-speed active-vision and high-speed, high-precision proximity sensors. The system processes the sensor data within 1 ms. The image of the vision sensor is preprocessed by an image processing PC and the PC calculates the center position of the object and sends the position data to dSPACE via Ethernet. The distance values of the two proximity sensors are sent to LPC1768(Microcontroller) via UART (1 Mbps). Further, LPC1768 sends the values to dSPACE via Ethernet.

is deformed by the impact force and vibration at contact, and grasping fails.

Therefore, we used a virtual damping controller to decrease the impact force and vibration at the time of contact. Fig. 8 shows the block diagram of the virtual damping control module. We defined the derivative of the distance detected by the proximity sensor as the virtual damping force and calculated the target torques of the joints based on the force in three steps. First, we calculated the virtual force using the following equation.

$$f_{\mathrm{damp_i}} = \frac{K_\mathrm{d}}{a - z^{-1}} \frac{k(z-1)}{T_\mathrm{s} z} d_\mathrm{i} \quad (i = 1, 2) \tag{6}$$

Here, $f_{\mathrm{damp_i}} (i = 1, 2)$ is the virtual damping force. $f_{\mathrm{damp_1}}$ is the force applied to the right fingertip, and $f_{\mathrm{damp_2}}$ is the force at the left fingertip. $K_\mathrm{d}$ is the gain of the virtual damping force, $k$ is the scaring factor of the discrete derivative, $T_\mathrm{s}$ is the sampling period (1 ms), and $a$ is a parameter of the low-pass filter. The gain $K_\mathrm{d}$ of the virtual damping control was experimentally determined. The distance output was filtered using a 1st-order IIR low-pass filter, and then the derivative was calculated and amplified using the gain. Second, to obtain the correct virtual damping force output, the first output on detecting the object was removed by using a filter (The details of the filter are shown in Fig. 8). Third, the target torques were calculated from the force using the principle of virtual work (Eq. 7).

$$\begin{bmatrix} \tau_{\mathrm{i}-1} \\ \tau_{\mathrm{i}-2} \end{bmatrix} = J_\mathrm{i}^\mathrm{T} \begin{bmatrix} f_{\mathrm{damp_i}} \\ 0 \end{bmatrix} \tag{7}$$

Here, $\tau_{\mathrm{i}-1}$ and $\tau_{\mathrm{i}-2}$ are target torques at flexion joint 1 and flexion joint 2 of finger $i$ (i = 1: right finger, 2: left finger). $J_\mathrm{i}^\mathrm{T}$ is the transposed Jacobian matrix of finger $i$.

Virtual damping control was executed continuously from the noncontact state to the contact state. Thus, the control module had two roles: 1) decreasing the approach speed of the finger (i.e., decreasing the impact force) and 2) reducing the vibrations of the fingertip at the contact.

## IV. High-Speed Catching Experiment

### A. System

The connection diagram of the system is shown in Fig. 9. The system consists of the high-speed robot hand, two active-vision sensors, two proximity sensors, an image processing PC, LPC1768 (Microcontroller), a hand driver, and dSPACE (main controller). As the data of each sensor are primarily processed in the PC or LPC1768, the amount of data processed by the main controller becomes small, and the control modules run fast.

The two proximity sensors detected the distance to the object and sent the data to LPC1768 via UART (1 Mbps). LPC1768 sent the data to dSPACE via Ethernet (UDP). The filtering process and calibration of the proximity sensor output are described in [3]. LPC1768 received the data from the two sensors and transmitted them to dSPACE via Ethernet (UDP).

The images of the vision sensors were processed by the image processing PC and sent to dSPACE via Ethernet. The sensor used was an IDP-Express R2000 (Photron), and the image size of the vision sensor was (512 × 512, color). We detected the object based on the color information. In marshmallow catching, white was extracted and the center position was calculated. On the other hand, in paper-balloon catching, yellow was extracted. In the image processing PC, first, a Bayer image was converted into a color image and an HSV image. Second, the HSV image was binarized with a threshold to extract the target object. Third, the center position of the object was calculated via image moment calculation. Finally, the center position values of two images were transmitted to dSPACE. In dSPACE, the positions of the object in 3D space were calculated from these values and the angles of the rotation stages of the vision sensor. The total processing time was less than 1 ms when using parallel processing and a lookup table.

### B. Marshmallow Catching

To determine the minimum fingertip opening width $W_{\mathrm{min}}$, the widths of ten marshmallows were measured, and the minimum one was approximately 23 mm and the maximum one was approximately 26.5 mm. In the experiment, the minimum one (23 mm) was used, and $W_{\mathrm{min}}$ is 30 mm (23 mm

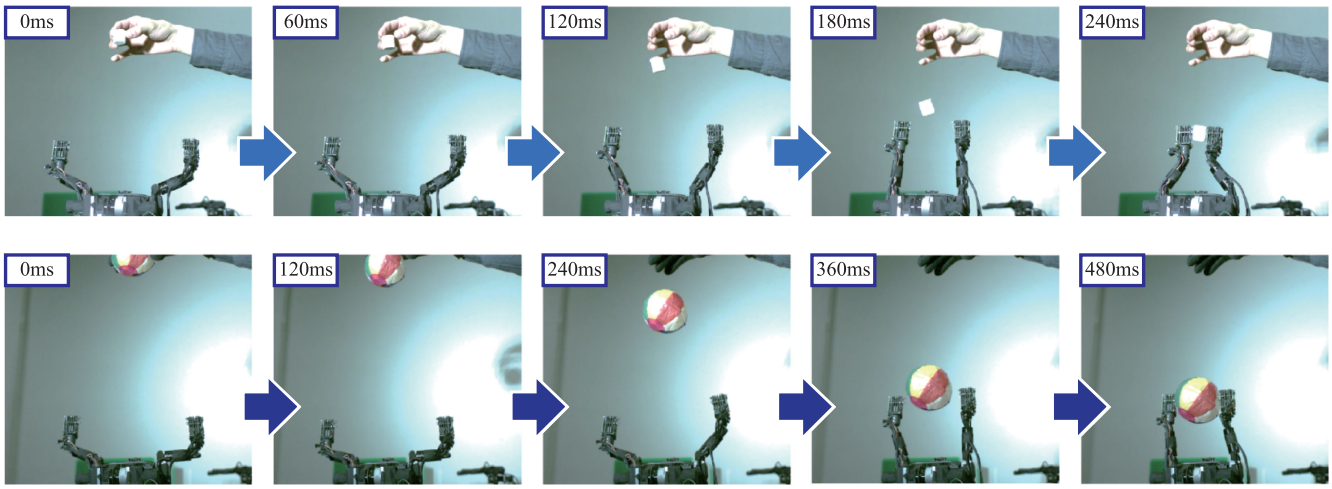


Fig. 10. Catching of the marshmallow and paper balloon. The marshmallow is dropped from a height of approximately 15 cm and the balloon is dropped from approximately 40 cm. The hand could catch the objects with small deformation.

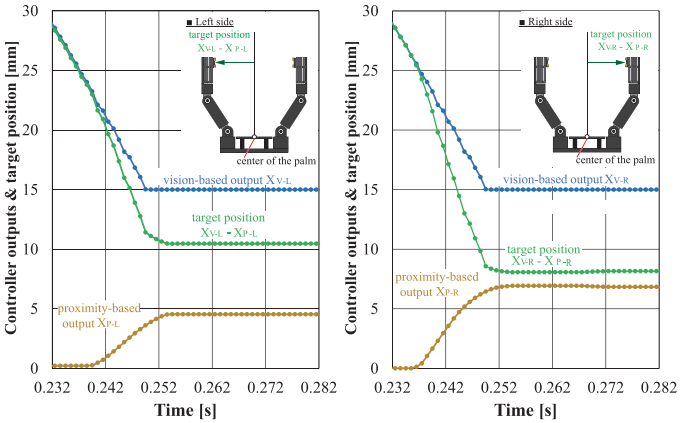


Fig. 11. Controller outputs and target position of the fingertip during the marshmallow catching process. The blue and yellow lines respectively show the vision-based controller output $X_{\mathrm{V-L}}$, $X_{\mathrm{V-R}}$ and proximity-based controller output $X_{\mathrm{P-L}}$, $X_{\mathrm{P-R}}$. The green line shows the sum of the two controller outputs (target position of the fingertip).

$+ 2 \times (26.5 - 23)$ set as the width to catch the marshmallow even when it was tilted).

A series of images showing marshmallow catching is shown in Fig. 10 (Upper). The hand could catch the marshmallow without crushing it.

Fig. 11 shows the controller outputs, which were calculated by the vision- and proximity-based controllers. In the figure, the left-side graph shows the outputs for the left finger, and the right-side graph shows the outputs for the right finger. The blue and yellow lines show the outputs of vision-based ($X_{\mathrm{V-L}}$, $X_{\mathrm{V-R}}$) and proximity-based ($X_{\mathrm{P-L}}$, $X_{\mathrm{P-R}}$) outputs. The green line shows the value of the vision-based output minus the proximity-based output. The green line indicates the target position of the fingertip. In Fig. 11, the vision-based outputs $X_{\mathrm{V-L}}$ (left side) and $X_{\mathrm{V-R}}$ (right side) reach the set width ($W_{\min}/2 = 15$ mm) at 0.249 s and do not change thereafter. On the other hand, the proximity-based output $X_{\mathrm{P-L}}$ increased thereafter and finally reached 4.54 mm for the left finger (For the right finger, the final value of $X_{\mathrm{P-R}}$ is 6.84 mm). Finally, the target position of the

TABLE I
AMOUNT OF DEFORMATION OF THE MARSHMALLOW AND PAPER BALLOON

| | Width of object *a* | Width between fingertips | | Amount of deformation ( *a* - *b* ) |
|---|---|---|---|---|
| | | target value | actual value *b* | |
| 1) marshmallow | ≒ 23mm | 18.62mm | 21.12mm | 1.88mm |
| 2) paper balloon | ≒ 87mm | 83.19mm | 83.60mm | 3.4mm |

left finger (green line) reached 10.46 mm, the target position of the right finger is 8.16 mm, and the width between fingertips is 18.62 mm as the sum of them. Table I 1) gives the width of the marshmallow (value $a$), target position, actual position (value $b$) between fingertips, and the amount of deformation (value $a - b$). Note that the amount of deformation includes the deformation of the marshmallow and rubber on the fingertip. The actual position was calculated using forward kinematics.

There was an error (2.5 mm) between the target and actual positions. This was because the fingertips were pushed back by the elasticity of the marshmallow and rubber on the fingertip. As the amount of deformation is 1.88 mm, the deformation of the marshmallow is 1.88 mm or less. The amount of deformation was almost the same as that in the case of a tactile-based control in static grasping conditions. The result of tactile-based control is described in Section V.

Fig. 12 shows the output $C_{\mathrm{out-L}}$, $C_{\mathrm{out-R}}$ of the proximity-based contact detectors. The left-side output $C_{\mathrm{out-L}}$ rises first from zero to one (Fig 12(a)), and then the right-side output $C_{\mathrm{out-R}}$ also rises (Fig 12(b)). However, the left-side output $C_{\mathrm{out-L}}$ falls from one to zero temporarily from 0.256 s to 0.273 s (Fig 12 (b)), because the distance output of the left proximity sensor $d_2$ became less than the minimum detection distance (2.85 mm). When the distance is smaller than this value, the distance output becomes "0". The reason why the distance output became less than minimum detection distance is that the rubber on the left finger (thickness 3 mm) was deformed by 0.15 mm or more by contact with the right finger. Thereafter, the output of the proximity-based controller $X_{\mathrm{P-R}}$ decreased

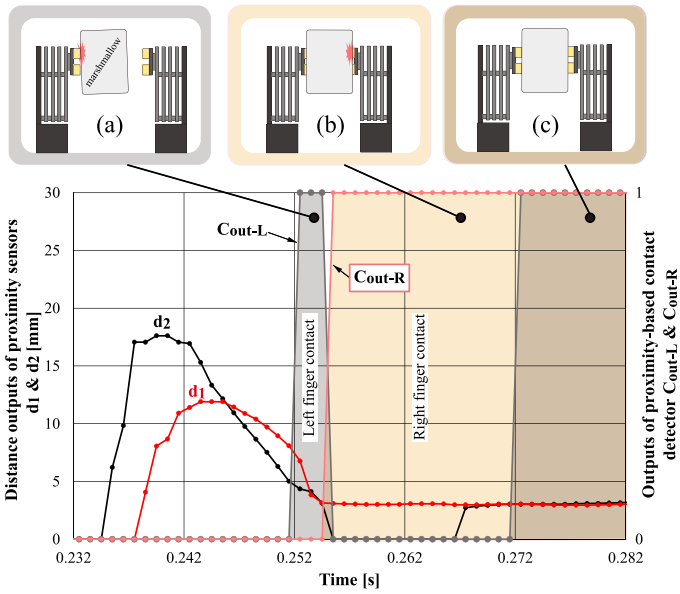


Fig. 12. Distance values of the proximity sensors and outputs of the proximity-based contact detector $C_{\mathrm{out-L}}$ (left side) and $C_{\mathrm{out-R}}$ (right side) in the marshmallow catching process. (a) $C_{\mathrm{out-L}}$ increases first from zero to one, and then the right-side output $C_{\mathrm{out-R}}$ also increases. (b) $C_{\mathrm{out-L}}$ changes temporarily from 0.256 s to 0.273 s because the distance output of the left sensor became less than the minimum detection distance by contact with the right finger. (c) The target position of the right finger changes, and $C_{\mathrm{out-L}}$ rises again at 0.273 s, and the motion of the fingers is stopped.

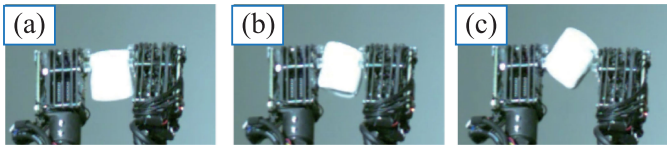


Fig. 13. Catching position of the marshmallow when the marshmallow was dropped three times. The catching position from the center of the palm is (a) −2.35 mm, (b) +1.78 mm, or (c) +5.28 mm. In case of (b) and (c), the posture of the marshmallow was tilted, but the hand could catch it with small deformation.

by 0.11 mm (from 6.93 to 6.84 mm) and the distance output of the left sensor became larger than 2.85 mm. The target position of the right finger changed in a direction away from the object, and the extent of rubber deformation on the left finger reduced to 0.15 mm or less. Finally, the left-side output of the proximity-based contact detector increased again at 0.273 s (Fig 12 (c)), and the grasping motion was stopped.

Thus, the hand caught the marshmallow using the following two steps: (1) The robot hand caught the object with a grasping force, and (2) the grasping position was adjusted so that the rubber deformation became minimum.

Fig. 13 shows the catching position of the marshmallow, when the marshmallow was dropped three times. The catching positions from the center of the palm −2.35 mm, +1.78 mm, and +5.28 mm were obtained by subtracting the values of the positions of the left finger and right finger. The positions were calculated using forward kinematics. As shown in Fig. 13, even if the marshmallow is tilted and the dropping position is slightly changed, the hand succeeded in grasping it without large deformation. When catching the marshmallow with a tilted posture,

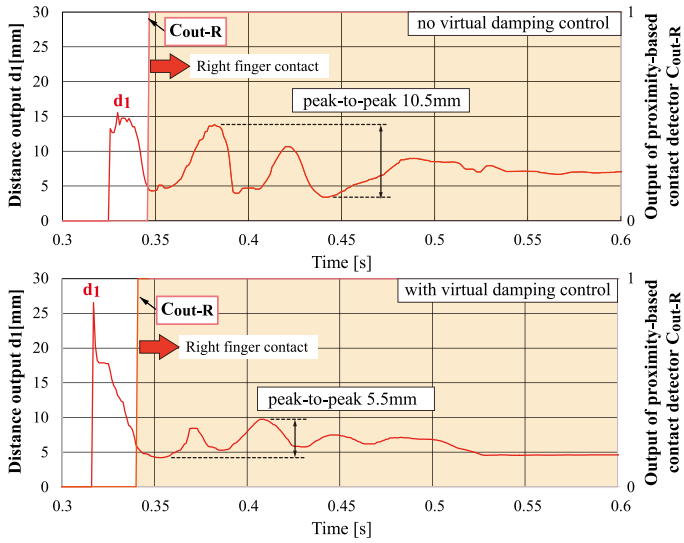


Fig. 14. Distance output of the proximity sensor in the paper-balloon catching process with no damping control (upper) and with damping control (lower).

the contact area becomes small and the hand tends to deform the surface considerably, although vision- and proximity-based grasping could catch it with small deformation (see video).

### C. Paper-Balloon Catching

In the paper-balloon catching experiment, we used all the four control modules. The sizes of the paper balloons were almost the same, and their shape was elliptical. The maximum diameter is approximately 100 mm, and the minimum diameter is approximately 87 mm. Therefore, we set the minimum fingertip opening width $W_{\mathrm{min}}$ to 100 mm.

Fig. 10 (lower) shows a series of images showing the paper-balloon catching process. The hand could catch the balloon without crushing it. Table I 2) gives the width of the paper balloon (value $a$), target position, and actual position (value $b$) between fingertips, and the amount of deformation (value $a - b$). The value $a - b$ indicates the amount of deformation of the paper balloon because the paper balloon is extremely soft compared to the rubber on the fingertip. The actual position was calculated using forward kinematics. From Table I 2), the deformation of the paper balloon is 3.40 mm. The vision- and proximity-based controllers enabled small-deformation catching compared to a tactile-based control described later (The result of tactile-based control is described in Section V).

Fig. 14 shows the right-side distance output in the case with no virtual damping control (upper figure) and with damping control (lower figure). From Fig. 14 (upper), without virtual damping control, the distance output vibrates at 10.5 mm from the minimum to the maximum distance between the fingertip and object after contact. On the other hand, from Fig. 14 (lower), the peak-to-peak vibration of the distance output was reduced to 5.5 mm. In the grasping experiments, when the virtual damping control was turned OFF, the fingertip vibrated after grasping. Therefore, the surface was deformed, and the hand failed to grasp in some

TABLE II
DEFORMATION AMOUNT IN THE CASE OF VISION- AND PROXIMITY-BASED CONTROL AND TACTILE-BASED CONTROL

| | Amount of deformation | |
|---|---|---|
| | vision- & proximity-based control | tactile-based control※1 |
| 1) marshmallow | 1.88mm | 1.85mm |
| 2) paper balloon | 3.40mm | 79.0mm |

※1static grasping

cases. On the other hand, when the virtual damping control was turned ON, the fingertip motion stopped smoothly.

## V. TACTILE-BASED GRASPING

We carried out grasping experiments using tactile-based control to compare the amount of deformation. In the experiment, we used a tactile and slip sensor (pressure-sensitive conductive-rubber type) [21]. The sensor can detect contact with a rigid object at a 5.4 g force. The object (marshmallow or paper balloon) was set at a fixed position between fingertips, as catching is difficult in tactile-based control. The fingertips were approached at a constant speed. The target speed is 134 mm/s (marshmallow) or 50.2 mm/s (paper balloon). These values were calculated from the change in the target position before contact in the vision- and proximity-based control. When the outputs of the two tactile sensors exceeded the threshold value, the fingertips were stopped.

Table II gives the amount of deformation of each object in the two different control methods.

Note that the value in Table II is only serves as a reference because the shape of the fingertip and softness are different. Moreover, the situation (dynamic catching or static grasping) is also different.

From Table II, the amounts of deformation of the marshmallow are almost the same. However, when the marshmallow was tilted, relatively large deformation occurred in the tactile-based control (see video). Moreover, the amount of deformation of the paper balloon with the vision- and proximity-based controls was more than 20 times smaller than the tactile-based control.

According to the obtained data, vision- and proximity-based control is more suitable for high-speed, small-deformation catching of soft objects than tactile-based control.

## VI. CONCLUSION

In this letter, we propose simple sensor feedback control modules based on high-speed vision and high-precision proximity sensing; we demonstrated high-speed and small-deformation catching of soft objects, i.e., a paper balloon and marshmallow. The combination of vision and proximity sensing enables high-speed (1 ms), precise (distance resolution $<50$ $\mu$m), and seamless sensing from a noncontact to a contact state without visual occlusion.

To realize high-speed operation of the control modules, we developed a 1 ms feedback system with Ethernet.

We confirmed that the robot hand could catch the marshmallow with small deformation even if the deformation model of the object was unknown and contact area was small. Furthermore, proximity-based virtual damping control became possible through accurate sensing of the distance. Virtual damping control was effective in catching an object whose surface can be easily deformed by a small impact force or vibration of the fingertips, such as a paper balloon.

## REFERENCES


[1] A. Namiki, Y. Imai, T. Senoo, and M. Ishikawa, "Dynamic active catching using a high-speed multifingered hand and a high-speed vision system," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2004, pp. 1849–1854.

[2] T. Senoo, A. Namiki, and M. Ishikawa, "Ball control in high-speed batting motion using hybrid trajectory generator," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2006, pp. 1762–1767.

[3] K. Koyama, M. Shimojo, T. Senoo, and M. Ishikawa, "High-speed high-precision proximity sensor for detection of tilt, distance and contact," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 3224–3231, Oct. 2018.

[4] C. Ma, C. Chang, T. Lin, and Y. J. Yang, "Highly sensitive tactile sensing array realized using a novel fabrication process with membrane filters," *J. Micro. Syst.*, vol. 24, no. 6, pp. 2062–2070, 2015.

[5] L. Jamone, L. Natale, G. Metta, and G. Sandini, "Highly sensitive soft tactile sensors for an anthropomorphic robotic hand," *IEEE Sens. J.*, vol. 15, no. 8, pp. 4226–4233, Aug. 2015.

[6] A. Rana, J. Roberge and V. Duchaine, "An improved soft dielectric for a highly sensitive capacitive tactile sensor," *IEEE Sens. J.*, vol. 16, no. 22, pp. 7853–7863, Nov. 2016.

[7] M. C. Bourne, *Food Texture and Viscosity: Concept and Measurement* (FOOD SCIENCE AND TECHNOLOGY A Series of Monographs). New York, NY, USA: Academic, 1981, p. 88, Table 7.

[8] T. Oka, N. Komura, and A. Namiki, "Ball juggling robot system controlled by high-speed vision," in *Proc. IEEE Int. Conf. Cyborg Bionic Syst.*, 2017, pp. 91–96.

[9] J. Kober, M. Glisson, and M. Mistry, "Playing catch and juggling with a humanoid robot," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2012, pp. 875–881.

[10] B. Bäuml, T. Wimbock, and G. Hirzinger, "Kinematically optimal catching a flying ball with a hand-arm-system," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 2592–2599.

[11] B. Bäuml *et al.*, "Catching flying balls and preparing coffee: Humanoid Rollin'Justin performs dynamic and sensitive tasks," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2011, pp. 3443–3444.

[12] S. Kim, A. Shukla, and A. Billard, "Catching objects in flight," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1049–1065, Oct. 2014.

[13] P. Cigliano, V. Lippiello, F. Ruggiero, and B. Siciliano, "Robotic ball catching with an eye-in-hand single-camera system," *IEEE Trans. Cont. Syst. Tech.*, vol. 23, no. 5, pp. 1657–1671, Sep. 2015.

[14] S. S. M. Salehian, M. Khoramshahi, and A. Billard, "A dynamical system approach for softly catching a flying object: Theory and experiment," *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 462–471, Apr. 2016.

[15] A. A. Bhole, J. Kumle, S. S. Groothuis, and R. Carloni, "Control of a variable stiffness joint for catching a moving object," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 4756–4761.

[16] A. Cretu, P. Payeur, and E. M. Petriu, "Soft object deformation monitoring and learning for model-based robotic hand manipulation," *IEEE Trans. Syst. Man, Cyber.*, vol. 42, pt. B, no. 3, pp. 740–753, Jun. 2012.

[17] T. Takahashi *et al.*, "Adaptive grasping by multi fingered hand with tactile sensor based on robust force and position control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2008, pp. 264–271.

[18] T. Nishimura, Y. Fujihira, R. Adachi, and T. Watanabe, "New condition for tofu stable grasping with fluid fingertips," in *Proc. IEEE Int. Conf. Automat. Sci. Eng.*, 2016, pp. 335–341.

[19] J. M. Krahn, F. Fabbro, and C. Menon, "A soft-touch gripper for grasping delicate objects," *IEEE/ASME Trans. Mech.*, vol. 22, no. 3, pp. 1276–1286, Jun. 2017.

[20] A. Yamaguchi and C. G. Atkeson, "Implementing tactile behaviors using fingervision," in *Proc. IEEE-RAS Int. Conf. Huma. Robot.*, 2017, pp. 241–248.

[21] S. Teshigawara *et al.*, "High sensitivity slip sensor using pressure conductive rubber for dexterous grasp and manipulationh," in *Proc. IEEE Int. Conf. Sens*, 2010, pp. 570–574.