Assistive Utilities with FingerVision for Learning Manipulation of Fragile Objects

Akihiko Yamaguchi

Abstract : This paper explores the use of vision-based tactile sensor FingerVision in learning manipulation of fragile objects. We consider to create utilities with FingerVision that assist learning manipulation of fragile objects. For example the grasp adaptation is a utility to adjust grasp to an object using slip feedback from FingerVision, which enables a robot to grasp a range of objects including deformable and fragile ones without accurate grasp plan. Other utilities presented in this paper are grasp failure detection, evaluation of grasp, emergency stop, and contact-event detection of grasped object. The benefits of these utilities include reducing the number of learning samples, and increasing the safety of exploration.
The accompanying video: `https://youtu.be/V0rwJRv2jdk`

1. Introduction

As well as vision, tactile sensation is considered to be important information in manipulating objects. In the context of robotic manipulation, a variety of tactile sensors and their applications have been researched. Recently applying artificial intelligence and machine learning including deep (reinforcement) learning to robot manipulation becomes popular. However many of them use vision but do not use tactile sensors. A reason would be a lack of *de facto* standard tactile sensors. Another reason is that proper strategies to use tactile sensors in learning manipulation have not been established.

This paper introduces a vision-based tactile sensor FingerVision [1] in the context of manipulation learning. FingerVision provides multimodal tactile sensation to robots including force and slip distributions, and shape, pose, and texture of object. It is applied to many types of robot behavior generation including manipulation [2, 3]. Notably, it is easy and inexpensive to manufacture. Since its manufacturing process and software are published as open source [4], anyone can reproduce it. Such an open-source tactile sensor could be a standard platform of learning manipulation with tactile sensors.

This paper demonstrates utilities with FingerVision that assist learning manipulation. A part of this work is inspired by palmar grasp reflex of infants (e.g. [5]). When an object is placed in the hand of an infant, the infant reflectively closes the fingers. Such a grasp reflex might play an important role in learning grasping objects. We explore to create preprogrammed behaviors to assist learning manipulation. Concretely, we introduce the following assistive utilities:

- Grasp adaptation: Adjusting grasp to an object with slip feedback from FingerVision. It enables a robot to grasp a range of objects including deformable and fragile ones without accurate grasp plan. This idea was initially explored in [3] that demonstrated the robot can grasp a wide range (around 30 kinds) of deformable and fragile objects without vision and parameter adjustment for each object.
- Grasp failure detection: Detecting grasp failure such as dropping an object.
- Evaluation of grasp: Learning manipulation needs to gather samples. In the framework of reinforcement learning, we use rewards. From a rich information of FingerVision, we create an evaluation of grasp.
- Emergency stop: When unexpectedly large force is applied to the fingers, we stop the motion of the robot. FingerVision is useful to detect such forces.
- Contact-event detection of grasped object: With a tactile sensor, the robot can sense force applied to the grasped object, and contact (collision) event on it. Such an event detection is useful to use tools, and place an object on a table.

The benefits of these utilities in learning manipulation would be: (1) reducing the number of samples, (2) increasing the safety of exploration, and (3) increasing the accuracy of evaluation.

Related Work

There are much work of robotic manipulation with tactile sensors. The basic one is grasping [6, 7, 8, 9] where tactile sensors are used to improve the grasp. In [10], a regrasping strategy was studied where the quality of grasp is estimated from the tactile sensing, and the regrasp is executed when the grasp is unstable. Tactile sensors are also used in other tasks such as opening a cap of bottle [6], rotating a cylinder [11], peg-in-hole [12], inserting a USB connector [13], cutting fruits [1], and in-hand manipulations [14, 15, 2].

In the context of manipulation learning with tactile sensors, contour-following control was learned with tactile sensors and reinforcement learning in [16] that was applied to close a ziplock bag. In [17], an active exploration was developed to search an object in unknown workspace and learn to discriminate the object. In [18], a tactile-based grasp learning strategy was studied where a touch based object localization and a tactile based regrasping methods were developed.

The remark of our work is proposing the practical use of tactile sensors especially FingerVision in manipulation learning. Although we demonstrate the use in learning grasping, the strategies are applicable to other tasks.

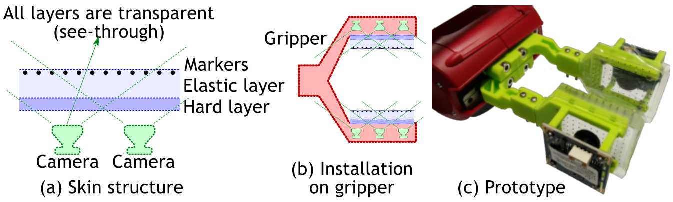

Fig. 1: Conceptual design of FingerVision (a, b) and its prototype installed on Baxter electric parallel gripper (c).

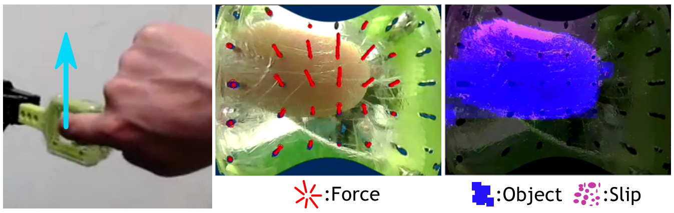

Fig. 2: Example of sensing when pushing the sensor by finger. Force distribution, slip, and object (finger) are detected.

2. FingerVision

FingerVision is a vision-based tactile sensor consisting of elastic and transparent skin made with silicone, frame made with 3D printed support and acrylic, and cameras. Fig. 1 shows the conceptual structure. Markers are placed on the surface of the skin for detecting skin deformation to estimate external force distribution.

The raw measurement from FingerVision is image sequence. We use several computer vision methods for estimating force distribution, slip distribution, and object pose. Estimating force distribution is achieved by tracking the markers whose movement tells us the deformation of the silicone skin. Estimating slip and object pose are done by directly analyzing the camera stream. We refer to such computer vision for nearby objects as *proximity vision*. Fig. 2 shows an example of marker tracking where the red bars highlight the marker movement, and detecting nearby-object and slip. Refer to [2] for the details.

3. Assistive Utilities with FingerVision

This section describes the assistive utilities with FingerVision in the context of manipulation learning.

3.1 Grasp Adaptation

Grasp adaptation is a control to adapt grasp to an unknown object where the gripper is controlled to avoid slip. In other words, this is a feedback control of slip. Grasping is considered as a control to prevent slip. With

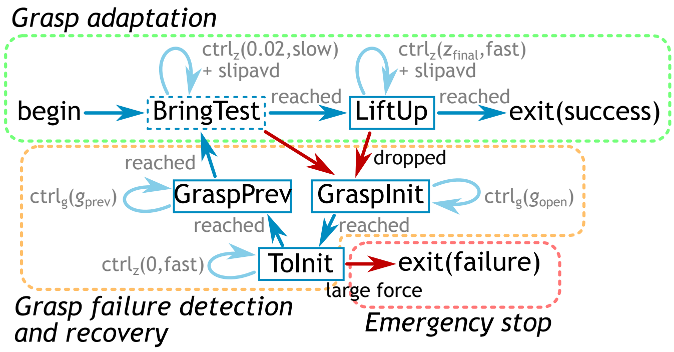

Fig. 3: Control scheme of the grasp adaptation. The detection of grasp failure and recovery, and emergency stop are unified.

a sensor that can detect slip, we can create a control strategy to prevent slip by adjusting the grasping force.

The sensitivity of slip of FingerVision is very high. Since it uses vision to detect slip, it does not depend on the contact force from the object. As the result, FingerVision can sense slip of lightweight objects such as origami arts.

Slip feedback control with FingerVision enables robots to adapt a grasp to a range of objects including lightweight, heavy, fragile, and deformable objects [3]. This means that even if a learning component outputs inaccurate grasp parameters, the slip feedback control may be able to correct the grasp. Thus the number of trials could be reduced.

The slip feedback control is implemented with a lifting-up motion. A robot tries to lift up an object with slip feedback control for the gripper. If the grasping force is not enough to hold the object, the slip feedback control adjusts the grasp. We refer to this controller as the grasp adaptation controller. Fig. 3 shows the control scheme of the grasp adaptation. First, the robot tries to bring up the object (`BringTest`) with the slip feedback control (`slipavd`). This motion is performed slowly so that the gripper can adapt the grasp to the object. Then the robot lifts it up to the final height. This motion is fast, while the slip feedback control is still active to adjust the grasp.

3.2 Grasp Failure Detection

Grasp failure detection is used to detect the fall of the grasped object. For this purpose, we use the object area on the images of FingerVision obtained by the proximity vision. If the area of the object becomes less than a ratio of the object area observed at the beginning of grasping, we consider the grasp has failed. Such a grasp failure happens when transporting an object, and trying to grasp an object.

In this paper, we combine the grasp failure detection and the grasp adaptation. If a grasp failure is detected during the grasp adaptation control (dropped the object when lifting it up), we make the robot try grasping again. Fig. 3 shows the control scheme of the grasp adaptation with the grasp failure detection. When the grasp failure is detected (`dropped`), the robot opens the gripper, moves back to the initial height, closes the gripper to the previous width, and tries the grasp adaptation again.

3.3 Evaluation of Grasp

In learning grasping, the evaluation of grasp is necessary as a part of the dataset. FingerVision can provide such an evaluation potentially in many ways. For example, (1) Evaluation by area: comparing the area of object before and after a grasp that indicates the amount of slip during grasping. (2) Alternatively we can use the position and orientation estimate. (3) Analyzing FingerVision images in more detail would give a damage to the object although we need advanced computer vision algorithms. In the implementation of this paper, we use (1).

3.4 Emergency Stop

In learning manipulation, a robot may take unexpected motions that produce large force; for example, pushing an object or a table too strongly with the finger. There are a variety of reasons why such motions are produced: exploration for improving the policy, error of visual perception, unexpected movement of the target or surrounding objects, and so on. Regardless the reason, such motions would damage the objects, the robot and the gripper, and the environment including humans.

FingerVision is useful to avoid such motions by detecting unexpected large force. Since a fisheye lens camera is used in FingerVision, it has a wide view including the fingertip. Even if FingerVision is facing horizontally, it can still sense the force applied from underneath. Of course the sensing range of FingerVision is limited; for example it cannot sense the opposite side of the finger surface. However we found that it can avoid the most of unexpected forces in learning grasping since such forces are produced only when moving the fingers down to the table or the target object. In these situations, sensing force from underneath works. Fig. 3 illustrates the example use of emergency stop that is integrated into the grasp adaptation control. Especially unexpected large force tends to be produced when moving the gripper back to the initial height (`ToInit`). When unexpected large force is detected, the movement is stopped.

3.5 Contact-Event Detection of Grasped Object

In learning manipulation, knowing the state of a grasped object is sometimes difficult. For example the pose of the object after graping is uncertain when it slips during grasping as shown in Fig. 4. Such uncertainty of the state would cause failure of manipulation. Even under such uncertainty, the robot can sense some information of the grasped object through FingerVision; e.g. external force applied to the grasped object, and contact-event (collision) of the grasped object. Such a contact-event detection of grasped object is useful to automate placing the object on a table. The uncertain if the object is touching the table or not is resolved with the contact-event detection. It can tell the robot when to release the object.

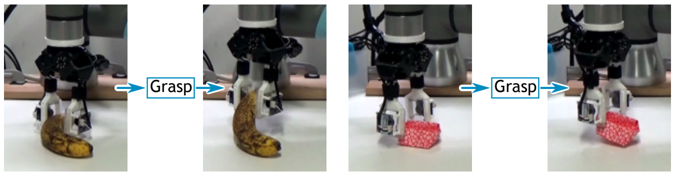

Fig. 4: Examples of uncertainty in poses of grasped objects that are caused by slippage during grasping.

In our implementation, we use the force and slip information from FingerVision to detect a contact-event. When grasping an object strongly, the force signal is effective to detect a force applied to the object. On the other side, the slip signal is useful when grasping a lightweight object such as an origami art, since the force used to grasp the object is too small to measure, but the slip is sensitively detected since it is obtained from image analysis.

4. Experiment

As a case study to using the assistive utilities with FingerVision, we explore a random pick-and-place of deformable and fragile objects.

4.1 Robot System Overview

We use a collaborative robot Universal Robots UR3 as a robot arm that has 6 degrees of freedom (DoF) and is driven by joint position or velocity commands. The robot accepts the joint velocity commands at 125 Hz. A 3D printed gripper actuated by a Dynamixel servo is mounted on the wrist of the robot that has 1 DoF. The servo is operated with the position control mode at 60 Hz, while the state is observed at 40 Hz. Two FingerVision sensors are attached on the fingers of the gripper. The data from FingerVision is processed at 30 Hz for obtaining force and slip distributions, and object area and pose. A wide-view camera is attached on the palm of the gripper that is used to find an object on a table. We do not use other external sensors. These devices are integrated with the control box of UR3, a Raspberry Pi 3B for streaming FingerVision camera data over Ethernet, another Raspberry Pi for streaming the palm camera, and a laptop PC with the Intel Core i7-8550U CPU. The laptop PC processes the FingerVision and palm camera data, computes the behavior described below,

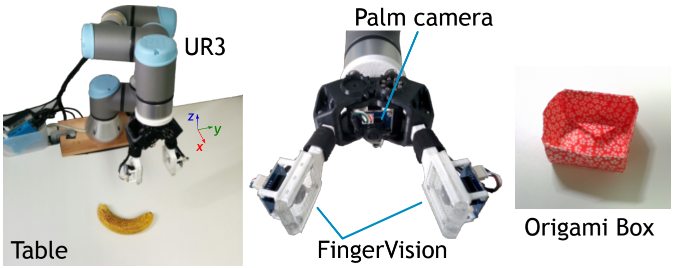

Fig. 5: Overview of the robot system.

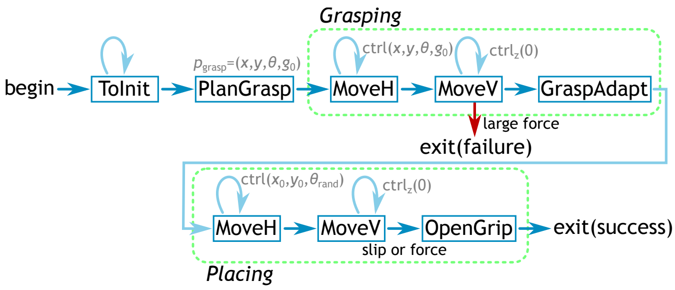

Fig. 6: Pick-and-place with the assistive utilities.

and sends control commands to the control box of UR3. Fig. 5 shows the overview of the robot system.

4.2 Pick-and-Place with Assistive Utilities

Fig. 6 shows the behavior design of pick-and-place where the assistive utilities with FingerVision are introduced. In the grasp adaptation (`GraspAdapt`), the grasp adaptation control, the grasp failure detection, and the emergency stop are used (cf. Fig. 3). In the `Grasping` part, another emergency stop is used at `MoveV` (moving vertically the gripper down to the table).

In the `Placing` part, the contact-event detection of the grasped object is used to determine the timing to open the gripper. Actually we use the slip and the force detection.

4.3 Grasp Planning

We assume a 2D grasping, i.e. the robot grasps an object always at the same height. The grasp parameters consist of 4 elements: the horizontal position (x, y), the orientation (rotation around z-axis), and the gripper width.

Planning grasping parameters is formulated as an optimization problem. This planner takes a reference grasp as an input. We optimize the grasp parameters close to the reference with avoiding the infeasibility and the failure estimated by the grasp estimation model. The infeasibility considers the collision and the inverse kinematics solvability. The grasp estimation model estimates the probability of grasp success implemented by neural networks.

The objective (cost) function takes an infinite value when the grasp is infeasible. Otherwise the cost function is given by:

$$\begin{aligned} L(\mathbf{p}_{\text{grasp}}) &= (1 + 10000 a_{\text{penetration}})^2 \\ &\quad + (100 p_{\text{failure}})^2 \\ &\quad + [0.01, 0.01, 0.001, 0] \cdot (\mathbf{p}_{\text{grasp}} - \mathbf{p}_{\text{ref}}) \qquad (1) \end{aligned}$$

where $\mathbf{p}_{\text{grasp}}$ denotes the grasp parameters, $\mathbf{p}_{\text{ref}}$ denotes a reference grasp, p_{failure} denotes a probability of failure computed by the neural networks, $a_{\text{penetration}}$ denotes the area of penetration, and dot $\cdot$ denotes inner product. The best grasp parameters are obtained by minimizing the cost function with respect to the grasp parameters $\mathbf{p}_{\text{grasp}}$. We use a gradient-free optimizer CMA-ES (Covariance Matrix Adaptation-Evolution Strategy) developed by Hansen [19].

4.4 Result

We repeat the pick-and-place behavior with random parameters to see if the proposed assistive utilities work. Two parameters are randomly decided: the reference grasp used in the grasp planning, and the orientation of placing the object. We use two types of objects: an origami box and bananas. During the experiment, the neural networks of the grasp estimation model are trained with the samples obtained through the execution.

Fig. 7 shows some representative scenes in an execution where the robot attempted to grasp an origami box. In the view of the palm camera, the contour detection of the target object (origami box) is shown. The robot fingers are masked. In the planning result, we can see the difference between the reference grasp generated randomly and the planned grasp. The reference grasp is infeasible (the fingertips are penetrating the object), while the planned grasp looks feasible. In the grasp adaptation, the robot adapted the grasp to the origami box where the slip feedback control with FingerVision was used. Note that without the grasp adaptation, the robot could not have picked up the object. The view of FingerVision after the grasp adaptation is also shown in the figure where are no slip points. In the placing phase, we can see that the robot stopped the motion before the fingertips reach the table. When FingerVision detected slip, the robot stopped the motion. The view of FingerVision at the slip detection is shown in the figure where we can see slip points.

Fig. 8 shows the change of success rate. The horizontal axis shows the number of runs (grasp attempts), and the vertical axis shows the success rate. Each marker is plotted when the training of the neural networks is done. In the training, all samples corrected before are used. The success rate is the rate of success samples between the current and the previous markers. Failures due to the detection of object contours are not counted as no actual grasp is attempted in such cases. Learning grasping bananas is conducted after that of origami

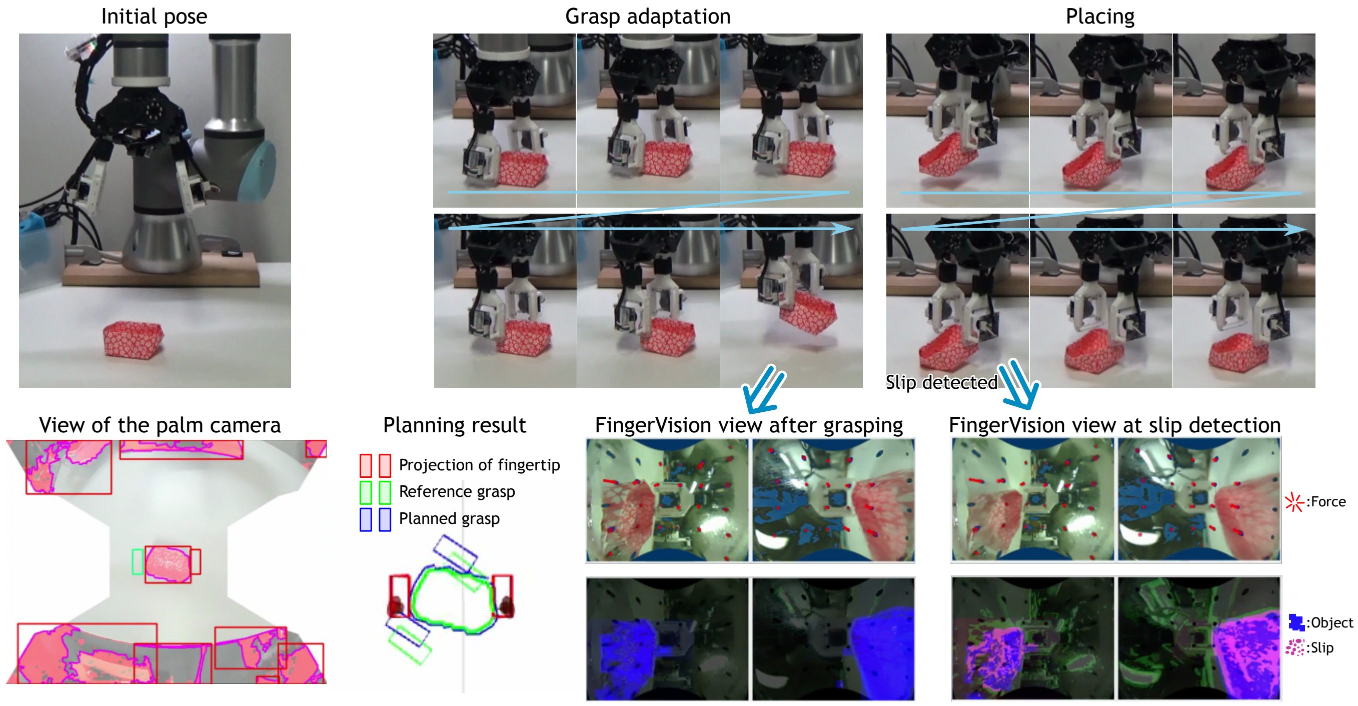

Fig. 7: An execution scenes of grasping origami box.

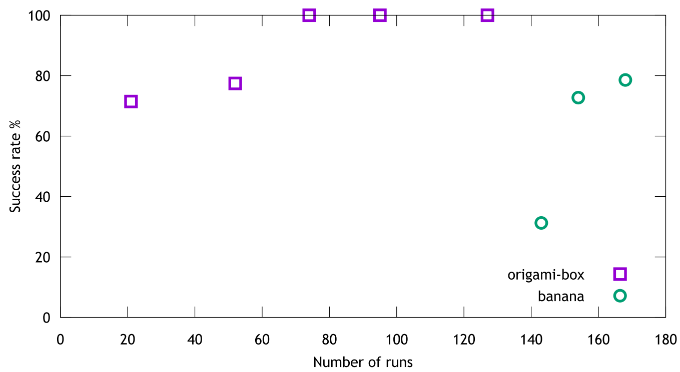

Fig. 8: Learning curve.

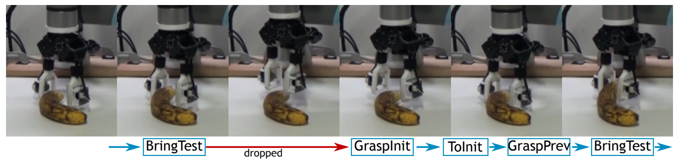

Fig. 9: Detection of drop and recovery to grasp the object again.

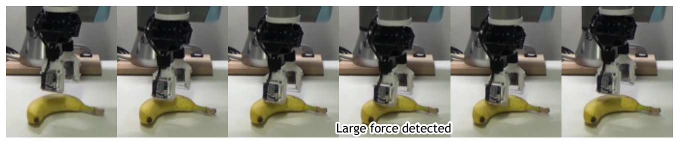

Fig. 10: Emergency stop in grasping a banana.

box where the samples from origami box are also used. The graph shows that in each case, the grasping performance is improved, and grasping bananas is more difficult than grasping origami box. Note that in our preliminary experiments, we tested to learn grasping bananas only with the samples obtained from grasping bananas. The performance was worse than that of grasping bananas in Fig. 8. Thus, the reason why the performance of grasping bananas is worse than that of origami box in Fig. 8 is not the sample sharing, but the difficulty of grasping bananas.

Through the experiments, we found that the assistive utilities with FingerVision worked in learning grasping. The grasp adaptation and the contact-event detection of grasped object are as shown in Fig. 7. Fig. 9 shows the case where the robot dropped the target object (banana) and recovered to grasp it again. The failure detection of grasp was used. Fig. 10 shows the emergency stop in grasping a banana. The banana was under a fingertip and they collided, which caused large force.

5. Conclusion

This paper explored the use of vision-based tactile sensor FingerVision in manipulation learning of robots. Especially we considered manipulating deformable and fragile objects. We created utilities with FingerVision that assist learning manipulation. The utilities explored in this paper are grasp adaptation, failure detection of grasp, evaluation of grasp, emergency stop, and contact-event detection of grasped object. We applied these assistive utilities to learning to grasp deformable and fragile objects (origami box and bananas). The experiments demonstrated that the developed assistive utilities are useful.

Acknowledgment

The author thanks to Mr Yoshihiro Shibata of ROBOTIS Japan for providing the 3D printed gripper with Dynamixel.

References

[1] A. Yamaguchi and C. G. Atkeson, "Combining finger vision and optical tactile sensing: Reducing and handling errors while cutting vegetables," in *the 16th IEEE-RAS International Conference on Humanoid Robots (Humanoids'16)*, 2016.

[2] ——, "Implementing tactile behaviors using FingerVision," in *the 17th IEEE-RAS International Conference on Humanoid Robots (Humanoids'17)*, 2017.

[3] ——, "Grasp adaptation control with finger vision: Verification with deformable and fragile objects," in *the 35th Annual Conference of the Robotics Society of Japan (RSJ2017)*, 2017, pp. 1L3–01.

[4] A. Yamaguchi, "FingerVision," http://akihikoy.net/p/fv.html, 2017.

[5] Y. Futagi, Y. Toribe, and Y. Suzuki, "The grasp reflex and moro reflex in infants: Hierarchy of primitive reflex responses," *International Journal of Pediatrics*, 2012.

[6] H. Yussof, J. Wada, and M. Ohka, "Sensorization of robotic hand using optical three-axis tactile sensor: Evaluation with grasping and twisting motions," *Journal of Computer Science*, vol. 6, no. 8, pp. 955–962, 2010.

[7] R. S. Johansson and J. R. Flanagan, "Coding and use of tactile signals from the fingertips in object manipulation tasks," *Nature Reviews Neuroscience*, vol. 10, no. 345, 2009.

[8] Z. Su, K. Hausman, Y. Chebotar, A. Molchanov, G. E. Loeb, G. S. Sukhatme, and S. Schaal, "Force estimation and slip detection/classification for grip control using a biomimetic tactile sensor," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, 2015, pp. 297–303.

[9] M. Kaboli, K. Yao, and G. Cheng, "Tactile-based manipulation of deformable objects with dynamic center of mass," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, 2016, pp. 752–757.

[10] Y. Chebotar, K. Hausman, O. Kroemer, G. Sukhatme, and S. Schaal, "Regrasping using tactile perception and supervised policy learning," in *AAAI Spring Symposium Series*, 2017.

[11] L. Cramphorn, B. Ward-Cherrier, and N. F. Lepora, "Tactile manipulation with biomimetic active touch," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 123–129.

[12] T. Hasegawa and K. Honda, "Detection and measurement of fingertip slip in multi-fingered precision manipulation with rolling contact," in *International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI 2001)*, 2001, pp. 43–48.

[13] R. Li, R. Platt, W. Yuan, A. ten Pas, N. Roscup, M. A. Srinivasan, and E. Adelson, "Localization and manipulation of small parts using GelSight tactile sensing," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3988–3993.

[14] V. A. Ho, T. Nagatani, A. Noda, and S. Hirai, "What can be inferred from a tactile arrayed sensor in autonomous in-hand manipulation?" in *2012 IEEE International Conference on Automation Science and Engineering*, 2012, pp. 461–468.

[15] H. van Hoof, T. Hermans, G. Neumann, and J. Peters, "Learning robot in-hand manipulation with tactile features," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, 2015, pp. 121–127.

[16] R. B. Hellman, C. Tekin, M. van der Schaar, and V. J. Santos, "Functional contour-following via haptic perception and reinforcement learning," *IEEE Transactions on Haptics*, vol. 11, no. 1, pp. 61–72, 2018.

[17] M. Kaboli, K. Yao, D. Feng, and G. Cheng, "Tactile-based active object discrimination and target object search in an unknown workspace," *Autonomous Robots*, 2018.

[18] A. Murali, Y. Li, D. Gandhi, and A. Gupta, "Learning to grasp without seeing," *ArXiv e-prints*, no. arXiv:1805.04201, 2018.

[19] N. Hansen, "The CMA evolution strategy: a comparing review," in *Towards a new evolutionary computation*. Springer, 2006, vol. 192, pp. 75–102.