2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Stable In-Grasp Manipulation with a Low-Cost Robot Hand by Using 3-Axis Tactile Sensors with a CNN

Satoshi Funabashi [1], Tomoki Isobe [1], Shun Ogasa [1], Tetsuya Ogata [2],
Alexander Schmitz [1], Tito Pradhono Tomo [1], and Shigeki Sugano [1]

Abstract— **The use of tactile information is one of the most important factors for achieving stable in-grasp manipulation. Especially with low-cost robotic hands that provide low-precision control, robust in-grasp manipulation is challenging. Abundant tactile information could provide the required feedback to achieve reliable in-grasp manipulation also in such cases. In this research, soft distributed 3-axis skin sensors ("uSkin") and 6-axis F/T (force/torque) sensors were mounted on each fingertip of an Allegro Hand to provide rich tactile information. These sensors yielded 78 measurements for each fingertip (72 measurements from the uSkin and 6 measurements from the 6-axis F/T sensor). However, such high-dimensional tactile information can be difficult to process because of the complex contact states between the grasped object and the fingertips. Therefore, a convolutional neural network (CNN) was employed to process the tactile information. In this paper, we explored the importance of the different sensors for achieving in-grasp manipulation. Successful in-grasp manipulation with untrained daily objects was achieved when both 3-axis uSkin and 6-axis F/T information was provided and when the information was processed using a CNN.**

I. INTRODUCTION

While multi-DOF robot hands already can accomplish various tasks that otherwise only humans can perform, the in-hand manipulation of everyday objects with such hands is still one of the most difficult challenges for the development of service and co-working robots. Specifically, manipulation with two fingers requires accurate control to avoid detaching the fingers from the grasped object, which would lead to dropping the object, and can include complicated contact states (e.g., rolling contact or slip). The current paper therefore focuses on subset of in-hand manipulation: 2-fingered in-grasp manipulation, in which the contacts between the object and the fingers are never broken.

Visual information is sometimes unreliable in determining contact states because of the occlusions caused by the hands. Therefore, tactile sensing can play an important role to ascertain the conditions between an object and fingers directly. For example, when the fingertips have an anthropomorphic shape and are covered with soft skin, the soft skin makes it easier to manipulate the object without dropping it, because it can deform along with the object, but controlling the hands can be difficult owing to the complex shape of the fingertips, in particular it can be difficult to achieve the desired goal configuration of the hand without applying too high forces. However, such in-grasp manipulation was achieved with abundant tactile information processed by deep neural networks [1].

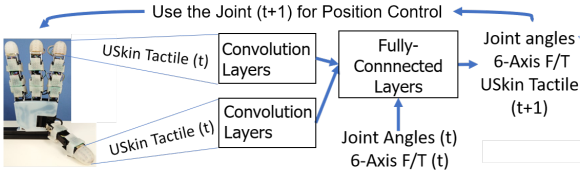

Use the Joint (t+1) for Position Control
USkin Tactile (t)
Convolution Layers
USkin Tactile (t)
Convolution Layers
Fully-Connnected Layers
Joint angles 6-Axis F/T USkin Tactile (t+1)
Joint Angles (t) 6-Axis F/T (t)

Fig. 1: Schematic of the proposed motion-generating method. CNNs process the uSkin tactile information from each fingertip. Fully-connected layers receive inputs from the joints, six-axis F/T sensors, and the features of uSkin tactile information. The outputs are the same as the inputs predicted for the next timestep, but only joint angles are used for robot hand control.

In our previous research on in-hand manipulation, we used the TWENDY-ONE Hand [1]. While it could demonstrate stable manipulation skills, the use of such a hand in industrial applications is limited by its high cost, caused by its highly sophisticated actuators and sensors. Moreover, maintaining such an elaborate robot is also challenging. Similarly, the multi-fingered Shadow Dexterous Hand can accomplish complex tasks [2], but costs about $100,000 and the tendon-driven actuators make the maintenance difficult. However, some more cost-effective hands are capable of achieving some simple in-hand manipulation tasks [3]. In the current study, we focus on the Allegro Hand, which is relatively low-cost ($15,000) among multi-fingered robot hands which have multiple actuators for each finger.

Low-priced hands such as the Allegro hand do not typically provide precise position or force control. To compensated for this and to achieve stable in-grasp manipulation similar to the capabilities of more expensive and elaborate hands, in the current paper, we focused on richer tactile feedback compared to our previous work in [1]. In particular, while the TWENDY-ONE hand has distributed 1-axis skin sensors (in addition to 6-axis F/T sensors in each fingertip), the current paper uses uSkin sensors [4][5], which provide distributed 3-axis measurements, yet are relatively cheap to produce. Like the sensors in [1], the uSkin sensors are soft.

However, dealing with higher-dimensional tactile informa-

This research was supported by the JSPS Grant-in-Aid No. 19H02116, No. 19H01130, the JST ACT-I Information and Future No. 50185, the Tateishi Science and Technology Foundation Research Grant (S), and the Research Institute for Science and Engineering, Waseda University.

[1] The authors are with the Faculty of Science and Engineering, Dept. of Modern Mechanical Engineering, Waseda University, Tokyo 169-8555, Japan. (e-mail: s_funabashi@sugano.mech.waseda.ac.jp)

[2] Tetsuya Ogata is with the Faculty of Science and Engineering, Dept. of Intermedia Art and Science, Waseda University, Tokyo 169-8555, Japan. (e-mail: ogata@waseda.jp)

978-1-7281-6211-9/20/$31.00 ©2020 IEEE

tion (in our case from distributed 3-axis sensors) provides a challenge in itself and the processing of high-dimensional tactile information has been less investigated than vision in particular. Nevertheless, some devices employ state-of-the art technology in the form of convolutional neural networks (CNNs) for tactile-based tasks such as hardness estimation from Gelsight videos [6], grasp success probability prediction from Gelsight images [7], or work by our own group with uSkin sensors on object recognition [8]. CNNs are well suited to extract features from spatially distributed sensors (from distributed tactile sensors in this case). Considering that the uSkin sensors provide more information than the ones in [1], for the current paper we also employed CNNs for each fingertip (Fig. 1) as one of our contributions. To the best of our knowledge, this is the first time that CNNs are used to process 3-axis tactile data and produce joint angles in an end-to-end fashion. Since, in our previous work, we already compared position control without tactile information to machine learning methods [1][9], in the current paper, we focus on the comparison only among different neural network architectures. In particular, in this paper the following aspects are studied

- Comparison between tactile sensors regarding their importance for successful in-grasp manipulation, i.e. either 3-axis or 1-axis tactile information from the uSkin sensors and/or 6-axis data from the F/T sensor is used as input for the neural network
- Analysis on how the CNN sees tactile information from the uSkin sensors.
- Generalization to untrained daily objects using a CNN and the uSkin and six-axis F/T sensors.

In previous work, our lab already showed that the triaxial tactile information provided by uSkin can be beneficial for object recognition tasks [8][10]. The contribution of the current paper is to investigate the usefulness of 3-axis skin sensor information and CNNs for processing abundant tactile information to achieve challenging in-grasp manipulation with a relatively low-cost (and low-precision control) robotic multi-DOF hand.

II. RELATED WORK

A. Sensors for In-Hand Manipulation

Various sensors have been used to aid in-hand manipulation. Visual sensors, such as commodity-level cameras, are a practical option. Some studies have incorporated visual information by using markers on grasped objects to capture the state and orientation of an object during manipulation [11][12]. However, putting markers on all manipulated objects is difficult to implement in practical situations. Furthermore, occlusions can occur. To avoid occlusions, a complicated experimental setting with a high number of cameras has to be used [2].

By contrast, tactile sensors can detect the contact states between objects and hands directly and are therefore also widely used. However, some tactile sensors cost more than the rest of the hands for which they are used [1][13]. Using such high-cost sensors would be prohibitive for low-cost systems. Some inexpensive optical sensors have been developed by 3D-printing [14][15] and can be used for performing stable in-grasp manipulation. However, a lot of space is required to accommodate such sensors with a camera, and therefore they cannot be used to cover multi-segment fingers, which would be beneficial for more dexterous in-hand manipulation.

For achieving more dexterous in-grasp manipulation tasks, soft skin and anthropomorphic curved fingertips can be beneficial [16]. Thus, our lab developed “uSkin” - distributed soft 3-axis tactile sensors that can be mounted on the surface of flat phalanges [4] as well as curved fingertips [5]. The distributed 3-axis information that the sensors provide could be beneficial, given that shear forces are important for stable in-grasp manipulation and in the detection of certain conditions such as slip. The detection of slip [17][18] or friction [19][20] has been used successfully for in-hand manipulation in previous work.

B. Control Architecture

When the tactile information increases, a control strategy that makes efficient use of such tactile information needs to be used. Contact modeling with external forces for in-hand object manipulation can be achieved [21][22]. Modeling is used for motion exploration based on tactile information [11][23], but this approach is applied to only elongated objects and requires complex object-specific modeling analysis. FEM is also used to handle soft objects [24], but not for dynamic in-grasp manipulation like changing an objects’ orientation. A combination of modeling and machine learning methods has been used for in-hand manipulation [25][26], but the generation of intricate manipulation is difficult to achieve because simple primitive motions are used. Overall, most studies focused on relatively simple in-hand manipulation or used conditions which made the task execution easier. This was necessary due to the difficulty of modeling grasping states with tactile information. Few robot hands are equipped with abundant tactile sensors and rich tactile information is also difficult to process using existing methods for in-grasp manipulation. Therefore, effective processing is a critical aspect of in-grasp manipulation and is still an open issue.

Currently, CNNs are used to process images obtained from cameras and to compress visual data to generate joint trajectories for robots [2]. For tactile sensors, CNNs are also used for recognition [7][27], and they utilize the results of the recognition for manipulation by modeling control [27]. CNNs are also used for predicting tactile force and directionality [28][29] for improving the robustness of in-hand manipulation. Therefore, CNNs could also be useful for processing tactile information for in-hand manipulation. To the best of our knowledge, the use of CNNs for processing 3-axis tactile information and directly generating joint angles in a robot hand for in-grasp manipulation as end-to-end learning has not yet been investigated.

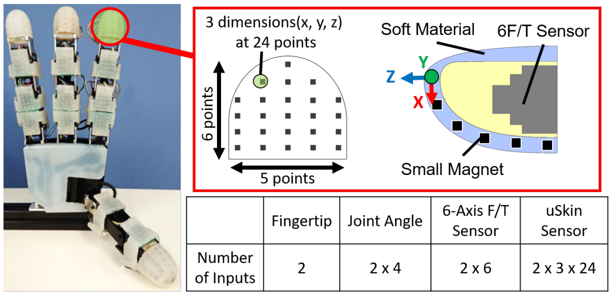
III. PROPOSED METHOD

A. Allegro Hand with Tactile Sensors

An Allegro Hand made by Wonik Robotics was used in this research because it is a low-cost multi-DOF robotic hand. Each finger has 4 DOFs (16 DOFs in total). The uSkin distributed tactile sensors, which can provide triaxial force measurements are mounted on the phalanges [4] and fingertips [5] of the Allegro Hand. Considering that this research focuses on in-grasp manipulation with fingertips, only the sensor information from the fingertips is described in Fig. 2. In addition to the 24 tri-axial uSkin sensors, each fingertip is instrumented with a six-axis F/T sensor. In this paper, we study 2-fingered in-grasp manipulation, and therefore the following measurements are used: 2 fingers * 4 joint angles + 2 fingertips * 6-axis F/T sensors + 2 fingertips * 24 uSkin sensor chips * 3 axes = 164 measurements.

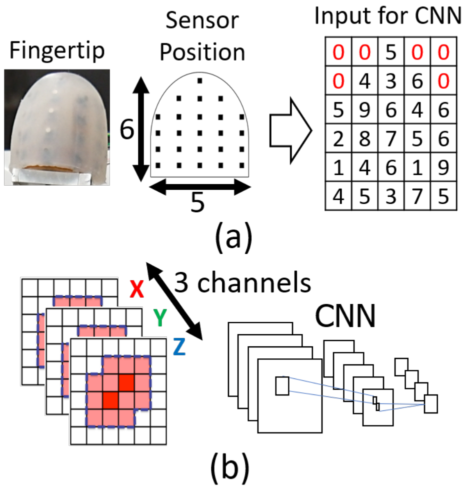
B. Convolutional Neural Network for Tactile Mapping

In this study, CNNs were used because a large amount of tactile information was provided, and such information could be difficult to process for basic MLPs or other machine learning methods. CNNs have demonstrated effectiveness in the areas of image recognition and tactile recognition [6][7][27] because they can extract features from spatially distributed sensors. They also use inputs with three channels for "RGB" because each pixel in an image has "RGB" information. Inputs from the uSkin were also entered into the CNNs in the same manner because each fingertip taxel provided "xyz" information (Fig. 3-(b)). The number of sensors on each fingertip was 24. According to the positions of the sensors, the shape of input maps for the fingertips was 6 * 5, and the red numbers "0" in Fig. 3 were entered for positions where sensors were not mounted on each fingertip, thus resulting in the construction of rectangular input maps for convoluting input maps by 2 * 2 (or larger sized) filters for the CNNs.



	Fingertip	Joint Angle	6-Axis F/T Sensor	uSkin Sensor
Number of Inputs	2	2 x 4	2 x 6	2 x 3 x 24

Fig. 2: Allegro Hand with tactile sensors. Each fingertip is covered with uSkin, and six-axis F/T sensors are mounted inside the fingertip. uSkin has 24 sensor points placed in a 6 * 5 configuration and each point has x-, y-, and z-axis tactile information. It is important to note that the six-axis F/T sensors were installed in accordance with the concept described in [1]. Z-axis represents normal force. X- and y-axis represent shear forces along with the shape of fingertips



0	0	5	0	0
0	4	3	6	0
5	9	6	4	6
2	8	7	5	6
1	4	6	1	9
4	5	3	7	5

Fig. 3: (a) Top row: mounted uSkin on a fingertip, black dots as positions of sensors, and simple input maps for the CNNs. Zero is colored in red and represents the position that lacks a sensor on the actual uSkin (the other numbers are arbitrary). (b) Bottom row: the input maps have three channels (x-, y-, and z-axes) associated with the CNN.

Fig. 4 shows the schematic of the proposed CNN, and Section IV-B and Table I list the detailed parameters. As described in Fig. 3, the input maps have a size of 6 * 5 * 3 from uSkin. The uSkin measurements pass through four convolutional layers, which are subsequently compressed by fully connected layers (FC layers "(uSkin)") so that the processed and compressed uSkin data has a dimensionality similar to the number of inputs from the other modalities (joint angles and six-axis F/T sensors). The compression of a high dimensional input modality before it is joined with other, lower dimensional sensor information, is a common technique. The joint angles and the six-axis F/T sensors are entered into the network in the first FC layer "(all)". Given that this research focused on geometric tactile information and on the manner in which it is processed, recurrent neural networks, including LSTMs, were not used, even though they are useful for processing time-series information for performing certain tasks, including in-hand manipulation.

Overall, the CNN uses as input the sensor readings from the current timestep and generates output for the next timestep. The size of the output is the same as that of the input (i.e. the measurements from joint angles, six-axis F/T sensors and uSkin sensors). The output is used for position control of the joints in each finger. This process is repeated resulting in generating in-grasp manipulation motion.

IV. EXPERIMENT DESIGN

A. Training Data

A data glove for controlling the TWENDY-ONE Hand was used in our previous work [1] to collect training data and to allow a teleoperator to control a robot hand with natural motion and human precision. However, such gloves are relatively expensive, and the parameters of the glove need to be calibrated for each person because of the differences between the hands of different people. Moreover, the Allegro

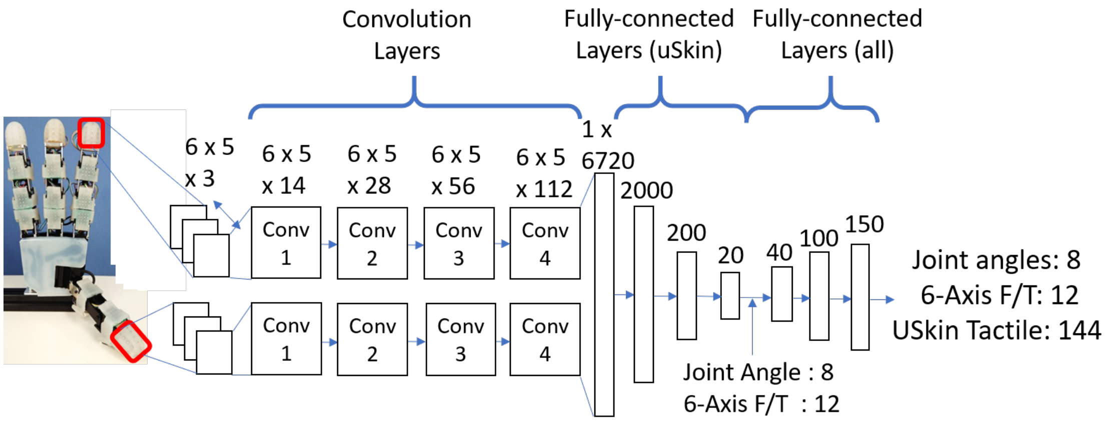


Fig. 4: The architecture of proposed network model. Tactile information from uSkin is input to a convolutional layer with the size of 6 * 5 * 3. Joint angles and 6-axis F/T sensors are input to a fully-connected layer because they do not have geometrical configuration unlike to uSkin. Although, outputs from the network include joint angles, 6-axis F/T sensors and uSkin measurements (all of them for the next time step respective to the input), only the joint angles are used to generate motions on the hand.

Hand used in this paper had a different joint configuration compared with that of human hands. Specifically, the Allegro Hand does not have abduction/adduction for all fingers, thus necessitating that increasing the distance between two fingertips in two dimensions, which commonly occurs in human motion, requires more than just the rotating of a joint in the Allegro hand. This condition makes precise teleoperation impossible in many cases. As a simple but powerful alternative method for obtaining training data, the experiments in this study employed posture interpolation control [30] because this method can generate in-grasp manipulation of multi-DOF hands easily [9]. The detailed formula of posture interpolation control is presented in [9]. The desired joint position could be achieved stably by posture interpolation control. In our previous work, we also discussed the limitations of interpolation control, namely that the start and end joint angles have to be defined for each object, that the manipulation is sensitive to the initial position of the object within the hand (if the objects are initially placed too far out of the center of the fingertips, the object can be dropped during the manipulation), and that high interaction forces can sometimes occur. For the selected training objects, we predefined the start and goal posture and the interpolation control could provide good motions of the desired in-grasp manipulation, and could be used for generating training data.

The target objects included a 40 mm diameter sphere and a 40 mm cylinder (Fig. 5-[a]). The objects used in this research are not deformable. The initial grasping positions of an object were determined by the teleoperator randomly to collect training data with as many initial grasping positions as possible so that the neural networks could be effectively trained with diverse motions. Fig. 5-(b) shows the target motions in our experiments: (b)-1 is a twisting motion with the object being rolled from the tip to the side of the index fingertip. The thumb is placed below the object in the beginning of the motion, but gradually moves to the side of the object, so that both the thumb and index finger are perpendicular to gravity. (b)-2 is a translating motion; both fingertips are perpendicular to gravity during the whole motion. Those motions were chosen because they are some of the more difficult in-grasp manipulation motions. They require a rolling contact and a change in the contact areas between the fingertips and the object during the motion. Therefore, those motions can be appropriate target motions for evaluating distributed tactile sensors and CNNs as they make spatial tactile changes during the motions. Moreover, the target motions are specific to anthropomorphic fingertips, and flat grippers, such as grippers in open-hand projects are not able to achieve these motions.

Each motion was executed 30 times successfully for each of the two objects, and 120 successful motions were recorded in total. Each of the 120 trials had 890 time steps (i.e., the elapsed time during the execution of the target motion was 8,900 milliseconds, and the sampling rate was 10 milliseconds).

Furthermore, the motion (b)-1 was the same as the one in [1], therefore the goal for the current proposed method could be set as the same success rate of in-grasp manipulation as in [1], even with a low-cost hand.

The neural networks were trained separately for each motion. 43,254 out of the 53,400 (890x30x2) available time steps were randomly chosen for training each network. Out of the remaining time steps, 4,806 were chosen randomly as the test set during optimizing the hyperparameters shown in Table I. Furthermore, the parameters in Table I were heuristically chosen on the basis of several (about 30 to 40) trials of in-grasp manipulations. Normalization was used to convert the values of all sensor measurements into values between -1 and 1 for inputting them into the neural networks. The CNNs for the two fingertips were trained separately (without weight sharing).

Information about the training data for the experiment with untrained objects is summarized in Section V-E.

B. Neural Network Settings

There are five architectures of neural networks (Table I) for the evaluation experiments described in Section V. Each input is abbreviated as follows; U3D is uSkin's 3-axis data, U1D is uSkin's z-axis data, 6F/T is 6-axis F/T data and JA is joint angles. The initializing method for weights in each neural network is He initialization because Relu was used as the activation function. Dropout was used in the 2nd Conv., 3rd Conv., 1st FC (uSkin), 2nd FC (uSkin), and 1st FC (all) with a rate of 0.5.

As mentioned in Section III-B, architecture I was used as our proposed network model for evaluating the effect of 3-axis tactile information and the CNN. As described in Section III-A, several zero measurements are added to the uSkin's inputs to achieve a rectangular input matrix for the 1st Conv. layer. As a result, the size of inputs for uSkin tactile information was 30 (24 [the number of sensor chips for one fingertip] + 6 [the number of "number 0"s for one fingertip]) * 3(forces axes) * 2 (number of fingertips) = 180 ([6 * 5 matrix] * 3 * 2). For the convolution layers, the filter size was 3 * 3, the stride was one, and the activation function was Relu. The size of outputs from the 4th Conv. layer was substantially larger than the number of inputs from the joint angles and the six-axis F/T sensors. Therefore, three fully connected layers (FC layer [uSkin]) were used to compress the outputs of the 4th Conv. layer. In the 1st FC layer (all), the outputs of the 3rd FC layer (uSkin), joint angles, and the six-axis F/T sensors were concatenated, thus resulting in a dimensionality of 40. The 3rd FC layer (all) provided outputs with a size of 164, in particular the next time step of the joint angles, the six-axis F/T sensors and uSkin sensors, so that the network could learn effectively. The next time step of the joint angles was used for controlling the fingers.

On the contrary, architecture II was prepared to investigate the effect of the 3-axis tactile information of architecture I. Consequently, architecture II was similar to architecture I. The only difference was the size of the input for the 1st Conv. Layer which was 6 * 5 * 1 (only normal tactile information [z-axis] was obtained from the uSkin because other distributed tactile sensors usually provide normal axis tactile measurements only).

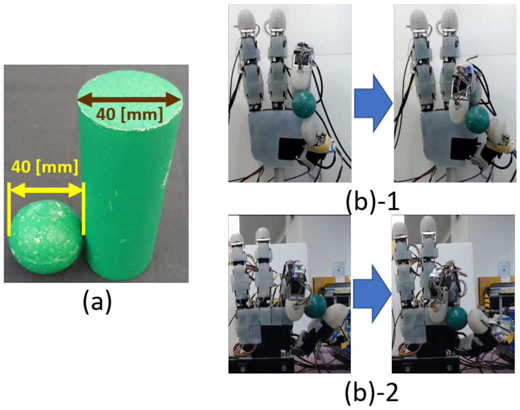


Fig. 5: (a) shows target objects made of Styrofoam used in the current paper. The objects are a sphere and a cylinder with 40mm of diameter as simple objects. (b)-1 and (b)-2 show the target motions.

Architecture III was set for evaluating in-grasp manipulation tasks when 6-axis F/T sensors are not used to check the effect of uSkin. Therefore, the size of the inputs was as follows (4 joint angles + 30 (24 [the number of sensor chips for one fingertip] + 6 [the number of "number 0"s for one fingertip]) * 3 (forces axes)) * 2 (number of fingertips) = 188.

Architecture IV had inputs that included only joint angles and six-axis F/T sensors and no information from uSkin. Therefore, the convolution layers were not necessary, and the network was an MLP with 4 layers. The size of the inputs was (4 joint angles + 6-axis F/T sensor) * 2 number of fingertips = 20.

Finally, architecture V had the same types of inputs (joint angles, six-axis F/T sensors, and uSkin) as those of architecture I, but the network was an MLP with 4 layers. By comparing architecture I (CNN) and V (standard MLP), the usefulness of the CNN can be tested.

Relu was used as an activation function for all layers, including convolution and FC layers, except for the output layer, which had no activation function. The outputs from the output layer had the same dimensionality as the inputs (joint angles, six-axis F/T sensors and uSkin), which were the predicted data for the next timestep. We assumed that the network acquires more useful features from the training data if it predicts the next time step of all the sensor data. However, only the joint angles were used for generating the robot hand's motion. The loss function was the mean squared error. Adam was used as the optimizer for all architectures with a learning rate of 0.00001, step size of 0.0001, first exponential decay rate of 0.9, second exponential decay rate of 0.999, and small value for numerical stability of 1e-08. The networks were trained with 43,254 samples (chosen randomly) for up to 35,000 epochs (the training was stopped earlier if the loss converged and if the model could already achieve success in-grasp manipulation) and the minibatch size was 100. All the networks are built with the TensorFlow library for Python and trained with Geforce GTX 1080 and RTX 2080 GPUs.

V. EVALUATION

A. Success Rate of Architectures

The usefulness of tactile sensors was investigated by comparing the success rates of in-grasp manipulation with architectures I to V. Success was defined as the case that the hand did not drop the object during the in-grasp manipulation and reached the desired grasping posture. In the beginning of each manipulation trial, the object was randomly placed on the fingertips. The hand was controlled by position control. Table II shows the success rates for in-grasp manipulation with the different architectures. For manipulation with the

TABLE I: Settings of Neural Networks

	Architecture	I	II	III	IV	V
	Network		CNN		FNN	
	All Inputs to Networks	200 U3D, 6F/T, JA	80 U1D, 6F/T, JA	188 U3D, JA	20 6F/T, JA	200 U3D, 6F/T, JA
	Inputs to 1st Conv.	180	60	180	-	-
1st Conv.	In/Out	3/14	1/14	3/14	-	-
	Filter Size		3, 3		-	-
	Stride		1, 1		-	-
	Activation		Relu		-	-
2nd Conv.	In/Out		14/28		-	-
	Filter Size		3, 3		-	-
	Stride		1, 1		-	-
	Activation		Relu		-	-
3rd Conv.	In/Out		28/56		-	-
	Filter Size		3, 3		-	-
	Stride		1, 1		-	-
	Activation		Relu		-	-
4th Conv.	In/Out		56/112		-	-
	Filter Size		3, 3		-	-
	Stride		1, 1		-	-
	Activation		Relu		-	-
	1st FC (uSkin)		2000		-	-
	2nd FC (uSkin)		200		-	-
	3rd FC (uSkin)		20		-	-
	1st FC (all inputs)	40		28	20	200
	2nd FC (all inputs)			100		
	3rd FC (all inputs)			150		
	Output	164	68	152	20	164
	Training Epochs			35000		
	Batch Size			100		

TABLE II: Achievement of the Final Posture

Architecture	Success Rate Twist		Success Rate Translate	
	Sphere	Cylinder	Sphere	Cylinder
I	8/10	7/10	10/10	8/10
II	4/10	0/10	0/10	0/10
III	0/10	0/10	9/10	8/10
IV	1/10	1/10	0/10	0/10
V	0/10	0/10	2/10	1/10

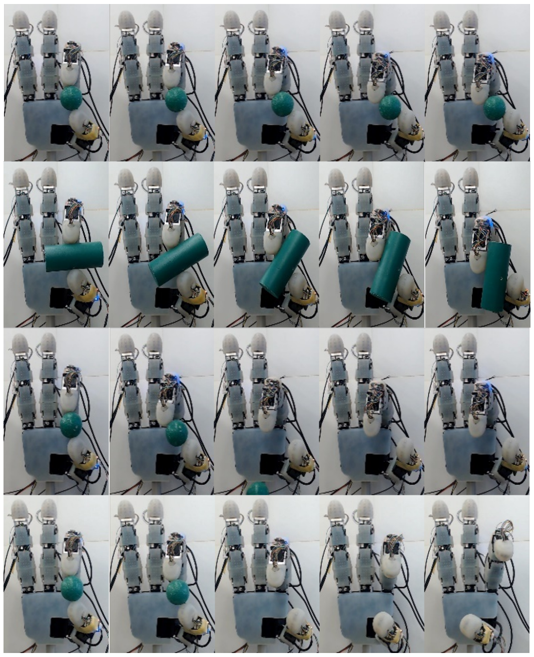


Fig. 6: Top and second rows: For the in-grasp manipulation with architecture I, the object was successfully manipulated. Third row: For the in-grasp manipulation with architecture II, the manipulated object was dropped on the way to the final grasping posture. Bottom row: in-grasp manipulation with architecture IV, the object dropped, and the final posture was wrong. (Grasping postures start from the left side.)

TABLE III: Error of the Final Posture (Translation)

	Sphere				Cylinder			
Pattern	FP [mm]		U3D		FP [mm]		U3D	
	Err	Var	Err x10^{-2}	Var x10^{-6}	Err	Var	Err x10^{-2}	Var x10^{-6}
I	11.2	0.19	4.0	4.75	11.54	2.40	2.3	0.15
II	132.7	0.58	4.7	$\fallingdotseq 0$	133.5	0.49	3.8	$\fallingdotseq 0$
III	13.4	1.14	4.4	1.2	11.31	3.07	2.8	17.3
IV	10.1	4.47	4.6	0.01	17.51	170.8	4.0	42.4
V	53.3	484	5.3	111.6	53.08	449.4	3.8	4.91

40 mm sphere, architecture I exhibited the highest success rate compared to the other architectures. Architecture III and V achieved no successful manipulation and architecture IV achieved only one for the twisting motion. Architecture III achieved successful translation motion, showing that 3-axis tactile information is effective. However, successful in-grasp manipulation with only 6-axis F/T sensors was achieved in our previous work [9]. The work in [9] implemented 4-fingered in-grasp manipulation, and we assume that four fingers provide more contact areas resulting in generating more stable manipulation while the manipulation task in the current paper was executed by two fingers only. Nevertheless, the six-axis F/T sensors are crucial for achieving stable in-grasp manipulation as the result of architecture III with twisting motion shows. The in-grasp manipulation of the cylinder with a diameter of 40 mm also shows a similar tendency in the results, with architecture I exhibiting the best result among the architectures. From this result, the usefulness of 3-axis tactile information from uSkin for in-grasp manipulation was shown, but six-axis F/T sensors are also important. As a comparison with architecture I, architecture V was also tested. However, it could perform almost no successful manipulation, which means that the MLP is not sufficient for processing the tactile information.

Examples of successful and unsuccessful in-grasp manipulations (twist motion) with different architectures are shown in Fig. 6.

B. Reachability of Architectures

In this section, the reached grasping postures are compared to the target posture for the translation motion. We use a metric from [31]. Table III shows the mean errors between the mean of desired final postures in the training set and the reached final postures by each architecture. Unlike in [31], tactile information plays an important role to achieve in-grasp manipulation in this study, and we therefore compared also the reached and desired grasping states, in particular the measurements of the 3-axis tactile (U3D) sensors were evaluated. The measurements of the tactile sensors were normalized to a range of 0 to 1. From the training set, we calculate for each 3-axis sensor the average length of the resultant force vector as the goal. These values are compared to the trials, the errors from all 3-axis sensors are summed up, and the average over all trials is reported. For the position of the fingertips the errors and the variances are similar except for architecture II and V. Architecture II generated a motion in which the fingertips moved in the opposite direction to the desired one every time. Therefore, architecture II has large errors in the fingertip positions. Regarding architecture V, this shows that the FNN cannot adequately handle the massive sensor information. It did not move the fingers to the desired final posture. Also, for the error of tactile information (U3D in Table III), architecture I and III produced smaller errors than architecture II, IV and V with both the sphere and cylinder. This result demonstrates that the proposed method using the CNN and 3-axis tactile information is capable of achieving more precise motions.

C. Analysis on Touch States and CNN

In Fig. 7 which tactile information the architecture I (CNN) regards as important is visualized using Grad-CAM++ [32] which provides a heatmap of calculated weights from the last convolution layer (4th Conv. in architecture I) corresponding to inputs (tactile arrays in this study). Unlike to classification problems, all the losses from the outputs of the architecture at each timestep were considered and calculated for generating the heatmap. Fig. 7 shows that the CNN gradually changes on to which part of the fingertip it focuses on during in-grasp manipulation. From this result, it was confirmed that the CNN effectively handled the tactile information from uSkin.

D. Generalization to Untrained Objects

Finally, architecture I was applied for in-grasp manipulation (twist motion) of untrained daily objects. Since the untrained objects had a variety of diameters (from 40 to 60 mm) and shapes (similar to spherical and cylindrical shapes) shown in Fig. 8 and Table IV, the training dataset for this evaluation experiment included spheres and cylinders with 40, 50 and 60 mm diameter. The goal was to evaluate the network's generalization skill. Thirty trials were collected for each training object and the collected data was randomly downsampled resulting in a training dataset with the size of 60,000 time steps for the CNN. The number of training epochs and the hyperparameters for the CNN were the same as the training setting of architecture I mentioned in Section IV-B. As shown in Table IV, all the daily objects were successfully manipulated, even better than the results in Table II. We assume that good success rates were achieved because the training dataset included manipulations with diverse objects, which made the network robust in terms of size and shape generalization. Furthermore, we assume that the softness and silicone cover of uSkin enhances the adaptability to different objects. The combination of human-mimetic fingertips with soft skin, rich tactile information and CNNs enabled successful in-grasp manipulation with a variety of untrained objects. In particular, the success rate for untrained everyday objects is almost the same as in our previous study with an accurate high-cost hand [1].

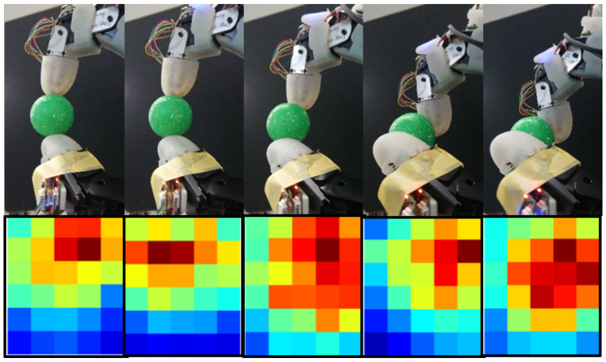


Fig. 7: The top row shows the in-grasp manipulation motion and the bottom row shows the corresponding heatmaps of Grad-CAM++ for the index finger as an example (the map arrays is same as Fig. 3 (a)). As the object moves on the fingertips, the focus of the the CNN also shifts. This shows how the CNN handles the tactile information and contributes to successful in-grasp manipulation.

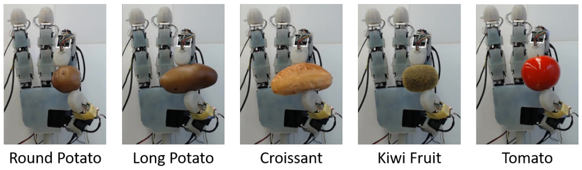


Fig. 8: Untrained daily objects used for in-grasp manipulation. They are made from Styrofoam, as are the objects used in training dataset. Therefore, the weight of each daily object is similar to that of the sphere and cylinder. Even though the texture for the daily objects is different for each object and similar to real foods (e.g., the tomato has a smooth and the kiwi a rough surface), which could make the in-grasp manipulation more difficult, they are all successfully manipulated.

TABLE IV: Achievement of the Final Posture with Untrained Objects

Objects	Size	Weight	Success Rate
Round Potato	47 mm	9 g	8/10
Long Potato	45 mm	4 g	9/10
Croissant	51 mm	8 g	8/10
Kiwi Fruit	45 mm	6 g	10/10
Tomato	52 mm	11 g	10/10

VI. CONCLUSIONS

This study was undertaken to investigate a method for in-grasp manipulation with a low-cost multi-DOF hand with 3-axis tactile sensors as well as 6-axis F/T sensors and CNNs. The mounted tactile sensors detected the grasping state, and a CNN was used to effectively process the information. As a result, even though the hand is low-cost and difficult to control precisely owing to its anthropomorphic shaped soft fingertips and backlash in the joints, the proposed method has the potential to generate successful in-grasp manipulation with abundant tactile information (especially adding shear forces) by employing a CNN which captures the change of the tactile information. Moreover, successful in-grasp manipulation with untrained daily objects was achieved.

The logical next step in our research is multi-fingered in-grasp manipulation tasks because the uSkin sensor can detect shear forces, which occur during multi-fingered in-grasp manipulation (e.g. slip and grasping from different orientations). Moreover, achieving several tasks with one network for avoiding re-training on each task can be a next challenge by utilizing one-hot vectors. [33].

References

[1] S. Funabashi, A. Schmitz, T. Sato, S. Somlor, and S. Sugano, "Versatile in-hand manipulation of objects with different sizes and shapes using neural networks," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, Nov 2018, pp. 1–9.

[2] OpenAI, "Learning dexterous in-hand manipulation," *CoRR*, vol. abs/1808.00177, 2018. [Online]. Available: http://arxiv.org/abs/1808.00177

[3] N. Rojas, R. R. Ma, and A. M. Dollar, "The gr2 gripper: An underactuated hand for open-loop in-hand planar manipulation," *IEEE Transactions on Robotics*, vol. 32, no. 3, pp. 763–770, June 2016.

[4] T. P. Tomo, W. K. Wong, A. Schmitz, H. Kristanto, A. Sarazin, L. Jamone, S. Somlor, and S. Sugano, "A modular, distributed, soft, 3-axis sensor system for robot hands," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, Nov 2016, pp. 454–460.

[5] T. P. Tomo, A. Schmitz, W. K. Wong, H. Kristanto, S. Somlor, J. Hwang, L. Jamone, and S. Sugano, "Covering a robot fingertip with uskin: A soft electronic skin with distributed 3-axis force sensitive elements for robot hands," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 124–131, Jan 2018.

[6] W. Yuan, C. Zhu, A. Owens, M. A. Srinivasan, and E. H. Adelson, "Shape-independent hardness estimation using deep learning and a gelsight tactile sensor," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 951–958.

[7] R. Calandra, A. Owens, D. Jayaraman, J. Lin, W. Yuan, J. Malik, E. H. Adelson, and S. Levine, "More Than a Feeling: Learning to Grasp and Regrasp using Vision and Touch," *ArXiv e-prints*, May 2018.

[8] S. Funabashi, G. Yang, A. Schmitz, A. Geier, and S. Sugano, "Morphology-specific convolutional neural networks for tactile object recognition with a multi-fingered hand," *2019 IEEE International Conference on Robotics and Automation (ICRA)*, May 2019, Accepted.

[9] S. Funabashi, A. Schmitz, S. Ogasa, and S. Shigeki, "Morphology-specific stepwise learning of in-hand manipulation with a four-fingered hand," *IEEE Transactions on Industrial Informatics*.

[10] S. Funabashi, S. Morikuni, A. Geier, A. Schmitz, S. Ogasa, T. P. Tomo, S. Somlor, and S. Sugano, "Object recognition through active sensing using a multi-fingered robot hand with 3d tactile sensors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep 2018 (Accepted).

[11] J. He, S. Pu, and J. Zhang, "Haptic and visual perception in in-hand manipulation system," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Dec 2015, pp. 303–308.

[12] P. Falco, A. Attawia, M. Saveriano, and D. Lee, "On policy learning robust to irreversible events: An application to robotic in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1482–1489, July 2018.

[13] J. Reinecke, A. Dietrich, F. Schmidt, and M. Chalon, "Experimental comparison of slip detection strategies by tactile sensing with the biotac®on the dlr hand arm system," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 2742–2748.

[14] B. Ward-Cherrier, N. Rojas, and N. F. Lepora, "Model-free precise in-hand manipulation with a 3d-printed tactile gripper," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2056–2063, Oct 2017.

[15] B. Ward-Cherrier, L. Cramphorn, and N. F. Lepora, "Tactile manipulation with a tacthumb integrated on the open-hand m2 gripper," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 169–175, Jan 2016.

[16] K. Or, A. Schmitz, S. Funabashi, M. Tomura, and S. Sugano, "Development of robotic fingertip morphology for enhanced manipulation stability," in *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, July 2016, pp. 25–30.

[17] F. Veiga, H. van Hoof, J. Peters, and T. Hermans, "Stabilizing novel objects by learning to predict tactile slip," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, pp. 5065–5072.

[18] M. Stachowsky, T. Hummel, M. Moussa, and H. A. Abdullah, "A slip detection and correction strategy for precision robot grasping," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2214–2226, Oct 2016.

[19] M. Costanzo, G. D. Maria, and C. Natale, "Slipping control algorithms for object manipulation with sensorized parallel grippers," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 7455–7461.

[20] F. E. V. B., Y. Karayiannidis, C. Smith, and D. Kragic, "Adaptive control for pivoting with visual and tactile feedback," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 399–406.

[21] N. Chavan-Dafle and A. Rodriguez, "Prehensile pushing: In-hand manipulation with push-primitives," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2015, pp. 6215–6222.

[22] N. C. Dafle, A. Rodriguez, R. Paolini, B. Tang, S. S. Srinivasa, M. Erdmann, M. T. Mason, I. Lundberg, H. Staab, and T. Fuhlbrigge, "Extrinsic dexterity: In-hand manipulation with external forces," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 1578–1585.

[23] J. Shi, J. Z. Woodruff, P. B. Umbanhowar, and K. M. Lynch, "Dynamic in-hand sliding manipulation," *IEEE Transactions on Robotics*, vol. 33, no. 4, pp. 778–795, Aug 2017.

[24] F. Ficuciello, A. Migliozzi, E. Coevoet, A. Petit, and C. Duriez, "Fem-based deformation control for dexterous manipulation of 3d soft objects," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 4007–4013.

[25] M. Liarokapis and A. M. Dollar, "Deriving dexterous, in-hand manipulation primitives for adaptive robot hands," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 1951–1958.

[26] M. V. Liarokapis and A. M. Dollar, "Learning task-specific models for dexterous, in-hand manipulation with simple, adaptive robot hands," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 2534–2541.

[27] W. Yuan, Y. Mo, S. Wang, and E. Adelson, "Active Clothing Material Perception using Tactile Sensing and Deep Learning," *ArXiv e-prints*, Nov. 2017.

[28] B. Sundaralingam, A. S. Lambert, A. Handa, B. Boots, T. Hermans, S. Birchfield, N. Ratliff, and D. Fox, "Robust learning of tactile force estimation through robot interaction," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9035–9042.

[29] K. Gutierrez and V. J. Santos, "Perception of tactile directionality via artificial fingerpad deformation and convolutional neural networks," *IEEE Transactions on Haptics*, pp. 1–1, 2020.

[30] K. Or, M. Tomura, A. Schmitz, S. Funabashi, and S. Sugano, "Interpolation control posture design for in-hand manipulation," in *2015 IEEE/SICE International Symposium on System Integration (SII)*, 2015, pp. 187–192.

[31] S. Cruciani, B. Sundaralingam, K. Hang, V. Kumar, T. Hermans, and D. Kragic, "Benchmarking in-hand manipulation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 588–595, 2020.

[32] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, March 2018, pp. 839–847.

[33] R. Rahmatizadeh, P. Abolghasemi, L. Bölöni, and S. Levine, "Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 3758–3765.