Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Real-time Liquid Pouring Motion Generation: End-to-End Sensorimotor Coordination for Unknown Liquid Dynamics Trained with Deep Neural Networks

Namiko Saito and Nguyen Ba Dai
Department of Modern Mechanical Engineering
Waseda University
Tokyo, Japan
n_saito@sugano.mech.waseda.ac.jp
badainguyen30@gmail.com

Tetsuya Ogata
Department of Intermedia Art and Science, Waseda University
National Institute of Advanced Industrial Science and Technology
Tokyo, Japan
ogata@waseda.jp

Hiroki Mori
Future Robotics Organization
Waseda University
Tokyo, Japan
mori@idr.ias.sci.waseda.ac.jp

Shigeki Sugano
Department of Modern Mechanical Engineering
Waseda University
Tokyo, Japan
sugano@waseda.jp

***Abstract*— We propose a sensorimotor dynamical system model for pouring unknown liquids. With our system, a robot holds and shakes a bottle to estimate the characteristics of the contained liquid, such as viscosity and fill level, without calculating to determine their parameters. Next, the robot pours a specified amount of the liquid into another container. The system needs to integrate information on the robot's actions, the liquids, the container, and the surrounding environment to perform the estimation and execute a continuous pouring motion using the same model. We use deep neural networks (DNN) to construct the system. The DNN model repeats prediction and execution of the actions to be taken in the next time step based on the input sensorimotor data, including camera images, force sensor data, and joint angles. At the same time, the DNN model acquires liquid characteristics in the internal state. We confirmed that the DNN model can control the robot to pour a desired amount of liquid with unknown viscosity and fill level.**

***Index Terms*— Neural networks, Long Short-Term Memory (LSTM) networks, Liquid characteristics estimation, Pouring**

I. INTRODUCTION

The demand for automatic liquid handling is increasing in industrial and research fields such as chemistry, biology, manufacturing as well as for home environments. Simple repetitive tasks, such as measuring liquid components, injecting or pouring liquids into containers, and stirring liquids, are often required for chemical and biological experiments and at manufacturing sites. If such tasks could be performed by robots, procedural errors and the need for people to handle dangerous chemicals could be reduced. In addition, everyday tasks involving liquids, such as serving drinks and preparing food, are among those most commonly envisioned for household robots [1].

Tasks involving liquids are difficult for robots because liquids have characteristics that, unlike solid objects, are complicated and motion-dependent. Wave propagation, separation, and changes in shape are generated in liquids in response to external stimuli. Thus, it is difficult to make traditional numerical models of liquids. Furthermore, pouring motions require consideration of the pouring angle and duration, both of which depend on characteristics that change in real time (e.g., fill level, viscosity, and density). Even if the target amount to be poured is the same, different motions are required if the characteristics of the liquid in the container are different. Thus, robots need to execute tasks in accordance with an estimation of a given liquid's characteristics.

The traditional method for estimating liquid characteristics is system identification, in which parameters of a liquid's characteristics are identified using observed data. However, previous studies were limited because separate models were required for estimating and manipulating liquids. These models were also difficult to use with unspecified types of liquids and with containers in unspecified environments.

Humans can shake a container and use proprioceptive feedback to perceive the characteristics of the liquid inside and handle the container even without knowing the specific parameter values of the liquid's characteristics. We take inspiration from human physiology and consider shaking by the robot as active perception. In other words, the robot observes how the liquid behaves depending on the shaking

*This work was supported in part by a JSPS Grant-in-Aid for Scientific Research (S) (No. 25220005), Grant-in-Aid for Scientific Research (A) (No. 19H01130), JST CREST (No. JPMJCR15E3), and the "Fundamental Study for Intelligent Machines to Coexist with Nature" program of the Research Institute for Science and Engineering, Waseda University, Japan.

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

movement and estimates its characteristics without determining the parameters.

We propose a liquid pouring model that shapes a dynamical system trained with DNN. The DNN model sequentially estimates liquid characteristics and generates a pouring motion command. At first, we control the robot to hold a liquid bottle and shake it to estimate the characteristics of the liquid contained inside. The process ensures the smooth transition of proper motion command and increases pouring accuracy. The motion commands generated by the model are described as the following discrete dynamical system.

$$m_{t+1} = f(m_t, m_{t-1} \cdots, s_t, s_{t-1} \cdots) \quad (1)$$

$$s_t, x_t = g(m_t, m_{t-1} \cdots, x_{t-1}, x_{t-2} \cdots) \quad (2)$$

Here, m is a motor command, s is observed sensor data, x is the robot's physical state, and t is a time step. The sampling time is fixed. The function g represents a physical system and the function f represents the sensorimotor coordination system, which is the target liquid pouring model trained to be acquired by the DNN in our context.

Our aim is to construct the dynamical system model using the DNN and to control the robot to pour a certain amount (e.g., 50 g) of liquid from a bottle to a saucer using two characteristics: liquid fill level and viscosity.

The DNN model comprises two modules, an image feature extraction module and a sensorimotor module. The image feature extraction module with a convolutional autoencoder (CAE) compresses the raw image and to generalize the untrained images. The sensorimotor module with long short-term memory (LSTM) networks plays a role in estimating liquid characteristics and generating pouring motions. It can generate next-step motion commands according to real-time sensorimotor feedback. After training, the model is tested on a NEXTAGE humanoid robot developed by Kawada Robotics [2]. For the evaluation, we use bottles of unknown liquids, for which viscosity and fill level are untrained. The results show the model estimates liquid characteristics and generates pouring motion commands.

Our contribution is as follows.

- The robot uses a single model to estimate liquid characteristics and generate pouring motion commands as a series of actions
- The model does not require a pre-designed environment and container, and the robot can handle unknown liquids

II. RELATED WORK

Previous research on robots using control theory allowed the robots to accurately perform liquid-related tasks as programmed by the experimenters. Kennedy et al. [3] calculated the arm angles necessary to pour specific amount of liquids using vision. Pan et al. [4] designed a model-based motion-planning system for fluid manipulation. Moriello et al. [5] built a system that suppresses sloshing while a robot handles

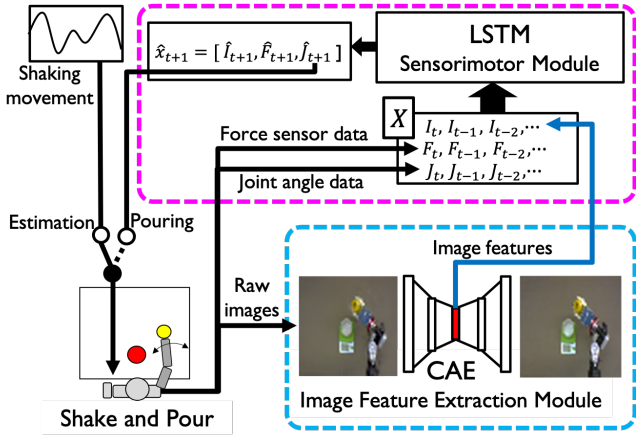

Fig. 1. Overall model comprising two modules. The image feature extraction module compresses the image data and extracts 10-dimensional image features, whereupon the sensorimotor module learns the image features, force data, and motor angles, thereby acquiring the liquid characteristics. The model can also predict suitable next-step arm joint angles.

liquids. However, the robots in these studies did not perform their tasks with unknown liquids efficiently. It was also difficult for the robots to change motions in response to different liquid characteristics.

We found three studies related to system identification that estimated liquid characteristics. Elbrechter et al. [6] proposed a method to recognize the viscosity of similar-looking liquids based on the visual information of the liquid's surface. Takamuku et al. [7] attempted to categorize liquids, papers, and solid objects contained in closed bottles based on the amplitude spectrum measured during shaking by a flexible robotic hand. Hara et al. [8] detected the level of a liquid using the visual information of the liquid's surface and its container. However, it was difficult to perform such estimations while the robot was executing its target task.

Schenck et al. [9] proposed a system in which a robot pours according to the amount of liquid. They estimated the amount of liquid in a container by using a hot liquid and computing the number of pixels in thermal images. Next, they controlled the robot to execute pouring tasks by inputting a parameter of the estimated amount of liquid into a proportional-integral-differential controller. Thus, the models for estimating and for pouring were separate.

The above studies considered only pre-determined pouring actions or specific liquids and containers. We construct a liquid-pouring model with which a robot can perform appropriate actions depending on the characteristics of any liquid.

III. LIQUID POURING DNN MODEL WITH LIQUID CHARACTERISTICS ESTIMATION

In this section, we describe our DNN model. The robot performs (i) shaking the container, which is the same motion each time in order to recognize the basic characteristics of the liquid and (ii) individual pouring motions that are appropriate

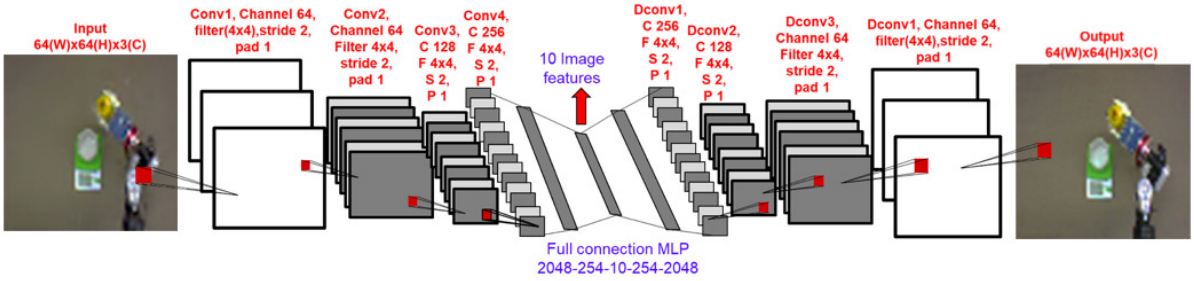

Fig. 2. Construction of the convolutional autoencoder (CAE). This network constitutes the image feature extraction module, which compresses 12,288-dimensional data [64 (width) × 64 (height) × 3 (channels)] and extracts 10-dimensional image features.

to the recognized liquid characteristics while simultaneously updating and refining the estimation.

A. Overview of Our Strategy

We need sensorimotor data to train function f in Eq. (1). We control the robot in order to have it experience shaking and pouring liquids of different viscosity in bottles with different fill levels. The motions were pre-programmed. We record sensorimotor data in the form of head-camera images, force-sensor data from the robot's right wrist, and the motor angles of its right arm.

After completing training with the recorded data, the model can predict the motor angle commands necessary for generating motions. First, the robot shakes the bottle to acquire sensor data s_t in Eq. (2). Next, the robot pours while adjusting the motor angle according to the characteristics of the liquid. In this case, the estimation of liquid characteristics and the generation of the pouring motion are realized as a series of actions.

B. Overview of Proposed Model

Figure 1 shows an overview of the proposed model. We conduct supervised learning with two DNN modules, namely the image feature extraction module and the sensorimotor module. The image feature extraction module extracts 10-dimensional image feature data from raw images. That is because the dimension of the raw image data is considerably larger than that of other data. By using this module, we can train the model on all the data, achieving a good balance at a low cost. The sensorimotor module learns to update the recognition of liquid characteristics. It also predicts next-step data from the current state and generates motion commands according to the characteristics at that time.

The input of the robot's motor commands is switched after completing the shaking motions. At first, the robot shakes according to the pre-programmed motor commands. After that, the robot pours according to the predicted commands generated by the sensorimotor module.

C. Image Feature Extraction Module

The image feature extraction module comprises a CAE [10] which has features of both an autoencoder (AE) and a convolutional neural network (CNN). An AE is trained to restore input data and can automatically extract the features of high-dimensional image data to low-dimensional data at the middle layer, which is called "latent space". A CNN is known to be an effective means of handling image data [11]. Consequently, a CAE can extract useful visual information with low-dimensional data as image features.

The construction of the CAE is shown in Fig. 2. For the activation function, we used the ReLU function. The weights are updated by back propagation using gradient descent [12]. Although the raw camera images initially have 12,288 dimensions [64 (width) × 64 (height) × 3 (channels)], the data is compressed to 10-dimensional image features. The picture on the right in Fig. 2 is the image restored, which is reconstructed to resemble the input picture on the left in Fig. 2. The model performs well at compressing the essential information to the latent space for subsequent restoration.

D. Sensorimotor Module

The sensorimotor module generates arm-joint angle commands after receiving the extracted 10-dimensional image features, the 6-dimensional force-sensor data, and the 6-dimensional joint angle data. This module comprises LSTM networks [13]. By using memory cells and a gating mechanism, the module determines which information should persist and when to read the memory state. This memory structure allows the model to learn and acquire the long-term and short-term features of time-series data.

In the following, X_t is the input vector and x_t is the sensorimotor data at step t, which integrates the image features (I_t), force-sensor data (F_t), and joint angles (J_t) at time step t. We combine T step data in the form

$$x_t = [I_t, F_t, J_t], \tag{3}$$

$$X_t = (x_t, x_{t-1}, x_{t-2}, \cdots, x_{t-T}), \tag{4}$$

so that the prediction is conducted using the data from the previous $(T + 1)$ steps. Time-series information can be learned without deepening the LSTM layer or increasing the number of neurons by inputting information multiple times. The networks can be calculated with a simple structure at a low cost.

Fig. 3. Bottle (left) and dishwashing liquids (middle) used in the experiments. Appearances of two types of liquid in a saucer (right).

The output vector $\hat{x}_t$ is calculated using LSTM. From the $\hat{x}_t$, we use $\hat{J}_t$ as motion commands, which are described as m_t in Eqs. (1) and (2). By repeating this process until the final step, the model uses the input visual and force information to generate the required motion commands.

The prediction error, E is calculated as

$$E = \sum_{t=T}^{T_f} (\hat{x}_t - x_t)^2, \tag{5}$$

and the data is trained to minimize E, where T_f is the final step. The weights are updated using back propagation through time and are optimized using the Adam optimization algorithm [14].

C_t is the internal state vector of the cells, that is, the "cell state." After training, LSTM recognizes the differences in the input of image and force data and clusters the information in the cell states. The input of image and force data depends on differences in how waves are created according to the characteristics of different liquids. We evaluate whether the robot correctly identifies the characteristics of the liquid by analyzing the internal values of the cell states.

IV. EXPERIMENTAL SETUP

A. System Design

We control the robot, NEXTAGE to generate movements with its right arm, which has 6 degrees of freedom. We move it remotely with a 3-dimensional mouse controller. The robot has two cameras attached to its head and we use the right-hand one. The camera takes pictures of size 64 (width) $\times$ 64 (height) $\times$ 3 (channels). We attached a Dyn Pick six-axis force sensor developed by Wacoh-Tech Inc. [15] on its right wrist and an EZGripper gripper developed by SAKE Robotics [16]. We record the joint angle, force data, and an image every 0.1 second at a sampling frequency of 10 Hz.

B. Object Used

The objective is to pour the target amount of liquid (50 g) into a white saucer. In identical yellow bottles, we prepare different levels of liquid for training (100 g, 150 g, and 175 g) and for testing (125 g). The bottles are closed, and the robot cannot see inside them. As shown in Fig. 3, the bottles are 150 mm tall and the saucers have a diameter of 100 mm. For training, we prepared two dishwashing liquids with different viscosities; one was watery (Enjoy Awa's Kitchen Detergent Fruit; JAN code: 4903367302922; viscosity: $\sim$ 100 mPa·s) and the other was more consistent (Frosch Dish Wash Blood Orange; JAN code: 4901670109993; viscosity: $\sim$ 230 mPa·s). In addition, we mix the two dishwashing liquids and use the mixture (viscosity: $\sim$ 160 mPa·s) for testing as a sample with untrained viscosity. We choose dishwashing liquids of similar color, as shown in Fig. 3, to preclude the possibility of discrimination based on color.

C. Task Design

We operate the robot to grasp the bottle and start moving it from a fixed initial position. A white saucer is placed in front of the robot to receive the liquid. The saucer is placed on a scale for weighing but the display on the scale is too small for the robot to read.

We set the time step of input T as 100. Therefore, $(T + 1) = 101$ time step data is combined as X_t and input to the sensorimotor module.

D. Training Dataset

We control the robot to execute the same shaking motion each time, as shown in Fig. 4. We set the shaking motions so that the liquid inside the bottle creates waves and allows the robot to sense the force difference depending on the characteristics of the liquid. The shaking motions continue for 12.0 seconds, resulting in 120 total time steps.

We direct different pouring motions for training according to the liquid level and viscosity. Different arm-joint commands for the roll axis of the wrist are shown in Fig. 6. The length of the pouring motion is 38.0 seconds (i.e., 380 steps). Thus, the final step T_f is $T_f = 120 + 380 = 500$.

In total, we collect 36 datasets (3 (levels) $\times$ 2 (viscosities) $\times$ 6 (data for each characteristic) with which to train the model. We train the CAE for 6,600 epochs and the LSTM for 5,100 epochs, and the error converges to a low value.

E. Evaluation of the System

We evaluate four aspects of the model as follows. We evaluate the actual poured amount based on the percentage error with respect to the target of 50 g as calculated by

$$\text{Percentage error} = \frac{(\text{Actual amount poured}) - 50}{50} \times 100. \tag{6}$$

1) Estimation of Liquid Characteristics: We evaluate the internal cell state of LSTM just after the shaking motion ($C_{t=120}$) by Principle Component Analysis (PCA) [17].

2) Motion Generation Ability of Our Model: We allow the robot to pour with our DNN model using trained bottles. We evaluate the accuracy of the amount poured compared with the target amount of 50 g.

3) Importance of Recognizing the Liquid Characteristics: We demonstrate the usefulness of our model by checking how much liquid is poured when the robot performs the pouring motion without our model but with motor angles programmed according to different characteristics.

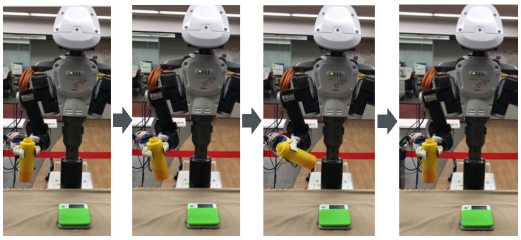

Fig. 4. The robot repeats the same shaking motion for a total of 12.0 seconds, during which 120 steps of data are recorded.

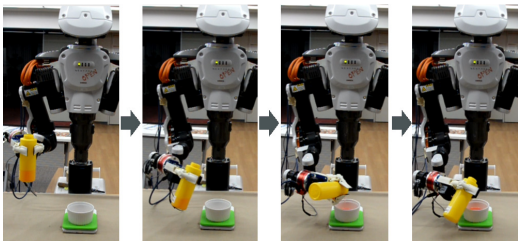

Fig. 5. The robot pours same amount of liquid (50 g) each time. The proper series of motor angles differs according to the characteristics of the liquid and takes less than 38.0 seconds (380 steps) to execute.

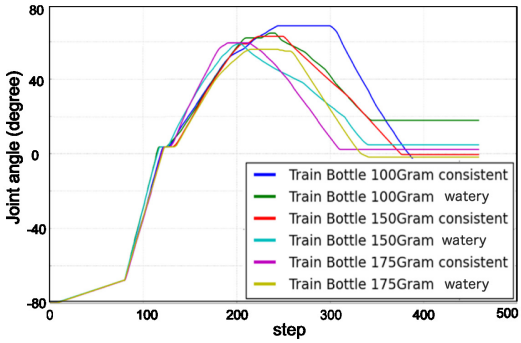

Fig. 6. Changes in motor angle of the wrist's roll axis necessary to pour 50 g of liquid according to level and viscosity, as programmed by the experimenter. The robot moves according to these changes and records data for training the model.

4) Motion Generation for an Untrained Liquid: To evaluate whether the robot can properly pour a liquid according to the characteristics of that liquid, we conduct pouring experiments six times for the following combinations of liquid level and viscosity: 125 g of watery soap, 125 g of consistent soap, and 125 g of mixed soap. We then evaluate the accuracy of the amount poured compared with the target amount of 50 g.

V. RESULTS

A. Estimation of Liquid Characteristics

Using PCA, the cell states were reduced to three dimensions, as shown in Fig. 7. We used different colors and shapes to plot the different characteristics of the liquids. The internal states are clustered according to liquid level and viscosity. PC1 represents viscosity; PC2 and PC3 represent fill level.

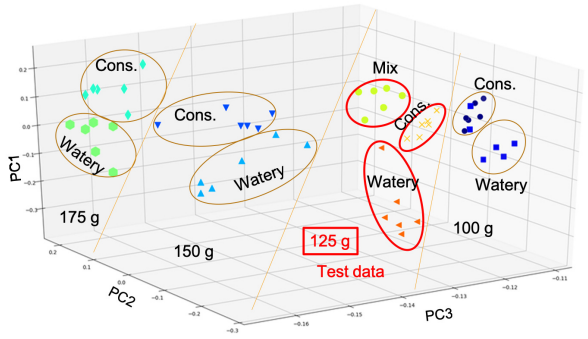

Fig. 7. PCA of the internal cell state. The color and shape of the plots depends on their characteristics; accordingly, the plots are clustered separately. Cons., consistent

TABLE I

AMOUNTS OF LIQUID POURED INTO THE SAUCER IN TRAINED SITUATIONS

Viscosity / Amount	Watery		Consistent	
	Average	Standard deviation	Average	Standard deviation
100g	48.2 *g* (-3.6 %)	1.1 *g*	48.6 *g* (-2.8 %)	0.7 *g*
150g	48.7 *g* (-2.6 %)	1.1 *g*	49.3 *g* (-1.4 %)	0.6 *g*
175g	51.2 *g* (+2.4 %)	0.8 *g*	51.8 g (+3.6 %)	0.5 *g*

B. Motion Generation Ability of Our Model

We controlled the robot to pour liquid with bottles trained using our model. The results are summarized in Table I. The largest percentage error was 3.6%. We confirmed that the model predicts proper motion commands.

C. Importance of Recognizing the Liquid Characteristics

We allowed the robot to generate pouring motion commands by using the motor angles programmed for the different characteristics. The results are summarized on the left side of table II. The robot poured an amount of liquid that was far from the target of 50 g (i.e., an error greater than 25%) in all combinations programmed for other characteristics. The results were nowhere close to the target, and thus demonstrate the importance of properly estimating liquid characteristics in order to accurately pour the target amount.

D. Motion Generation for Untrained Liquid

We also used our model to conduct experiments using liquids with untrained level and viscosity. The robot generated pouring motions with the appropriate angles and length of time and poured an amount close to the target of 50 g. As shown in Fig. 8, the robot poured smoothly and never spilled liquid outside of the saucer. Table II summarizes how many grams were poured on average and the percentage error for each liquid with untrained viscosity. The percentage errors for 125 g of watery, consistent, and mixed soap using our

TABLE II

AMOUNTS OF LIQUID POURED INTO THE SAUCER FROM THE BOTTLE CONTAINING LIQUID WITH UNTRAINED CHARACTERISTICS.

Angle / Test liquid	Designed for				Generated by the model		
	100 g Watery	150 g Watery	100 g Consistent	150 g Consistent	Average	Standard deviation	Largest error
125 g Watery	77.1 *g* (+54.2 %)	36.6 *g* (-26.8 %)	93.2 *g* (+86.4 %)	75.0 *g* (+50.0 %)	47.0 *g* (-6.0 %)	2.5 *g*	44.8 *g* (-10.4 %)
125 g Consistent	33.5 *g* (-33.0 %)	3.7 *g* (-92.6 %)	66.9 *g* (+33.8 %)	35.5 *g* (-29.0 %)	53.9 *g* (+7.8 %)	1.2*g*	56.1 *g* (+12.2 %)
125 g Mix	36.2 *g* (-27.6 %)	10.7 *g* (-78.6 %)	86.6 *g* (+73.2 %)	37.4 *g* (-25.2 %)	53.4 *g* (+6.8 %)	0.9 *g*	55.5 *g* (+11.0 %)

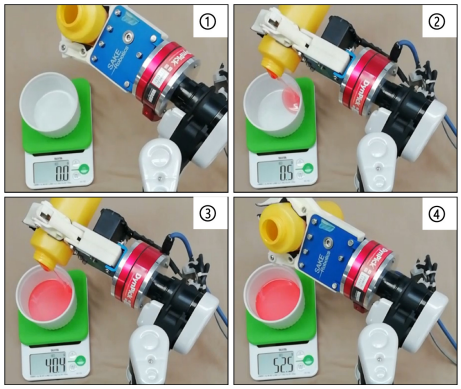

Fig. 8. Result of the generated pouring motion using a bottle containing 125 g of mixed soap. The robot was able to pour with the appropriate angles and length of time according to the characteristics of the liquid.

model were -6.0%, $+7.8\%$, and $+6.8\%$, respectively. These errors are much smaller than those using the programmed motor angles. In addition, the standard deviation was small, confirming that the DNN model generated motion commands with high reproducibility.

VI. CONCLUSION

We constructed a sensorimotor dynamical system for pouring unknown liquids. From the results, we showed our model accurately estimates liquid characteristics without calculating specific parameter values, and accordingly generates appropriate pouring motion commands. Our model is highly generalizable and enables the robot to handle liquids that has not previously been trained on. The robot pours the target amount of liquid with a percentage error of typically less than 12.2%. The model is useful in fields where small quantities of many different types of liquids are handled.

We showed our model can be applied to tasks that require adjusting movements in accordance with changing characteristics. Such a model must be able to deal not only with liquids but also with flexible objects. In future research, we will apply the DNN model to other, more complex materials. In addition, the model should be improved so that the target amount of poured liquid (50 g) can be changed. We will update the model to learn the target amount as well.

REFERENCES

[1] Kimitoshi Yamazaki, Ryohei Ueda, Shunichi Nozawa, Masayuki Inaba et. al. “Home-Assistant Robot for an Aging Society,” Proceedings of the IEEE, Vol. 100, No. 8, pp. 2429–2441, 2012.

[2] Kawada Robotics, ”Next generation industrial robot Nextage,” http://nextage.kawada.jp/

[3] Monroe Kennedy, Kendall Queen, Dinesh Thakur, Kostas Daniilidis and Vijay Kumar, ”Precise dispensing of liquids using visual feedback,” in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1260–1266, 2017.

[4] Zherong Pan and Dinesh Manocha, ”Motion planning for fluid manipulation using simplified dynamics,” in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4224–4231, 2016.

[5] Lorenzo Moriello, Luigi Biagiotti, Claudio Melchiorri and Andrea Paoli, ”Control of liquid handling robotic systems: A feed-forward approach to suppress sloshing,” in Proceedings of IEEE International Conference on Robotics and Automation (ICRA), pp. 4286–4291, 2017.

[6] Christof Elbrechter, Jonathan Maycock, Robert Haschke: ”Discriminating Liquids Using a Robotic Kitchen Assistant”, in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015.

[7] Shinya Takamuku, Koh Hosoda and Minoru Asada, “Object Category Acquisition by Dynamic Touch,” Advanced Robotics, Vol. 22, pp. 1143–1154, 2008.

[8] Yoshitaka Hara, Fuhito Honda, Takashi Tsuboguchi, “Detection of Liquids in Cups Based on the Refraction of Light with a Depth Camera Using Triangulation,” in Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (ICRA), 2014.

[9] Connor Schenck, Dieter Fox: ”Visual Closed-Loop Control for Pouring Liquids,” in Proceedings of IEEE International Conference on Robotics and Automation (IROS), 2017.

[10] Geoffrey E. Hinton and R. R. Salakhutdinov, ”Reducing the dimensionality of data with neural networks,” Science, vol. 313, no. July, pp. 504-507, 2006.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton, ”ImageNet classification with deep convolutional neural networks,” in NIPS’12 Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1, pp. 1097–1105, 2012.

[12] YannLecun, Leon Bottou, Yoshua Bengio, Patrick Haffner, “Gradient-Based Learning Applied to Document Recognition,” Proceeding of the IEEE , Vol. 86, pp. 2278-2324 1998.

[13] Sepp Hochreiter, Jurgen Schmidhuber, “Long Short-Term Memory,” Neural Computation, Vol. 8, pp. 1735–1789, 1997.

[14] Paul J. Werbos, “Backpropagation Through Time: What It Does and How to Do It,” Proceedings of IEEE, vol.78, No.10, pp.1550–1560, 1990.

[15] Force sensors (WACOH-TECH), https://wacoh-tech.com/en/

[16] SAKE Robotics: Robot Grippers, https://sakerobotics.com/

[17] Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik, “A training algorithm for optimal margin classifiers,” in Proceeding of he fifth annual workshop on Computational learning theory, pp. 144–152, 1992.