

# Motion Generation by Learning Relationship between Object Shapes and Human Motions

Tokuo Tsuji<sup>\*1</sup>, Sho Tajima<sup>2</sup>, Yosuke Suzuki<sup>1</sup>, Tetsuyou Watanabe<sup>1</sup>,  
Shoko Miyauchi<sup>3</sup>, Ken'ichi Morooka<sup>3</sup>, Kensuke Harada<sup>4</sup>, and Hiroaki Seki<sup>1</sup>

<sup>1</sup>*Institute of Science and Engineering, Kanazawa University,  
Kakuma-machi, Kanazawa-shi, Ishikawa 920-1192, Japan.*

<sup>2</sup>*Graduate School of Natural Science and Technology, Kanazawa University,  
Kakuma-machi, Kanazawa-shi, Ishikawa, 920-1192, Japan.*

<sup>3</sup>*Faculty of Information Science and Electrical Engineering, Kyushu University,  
744, Motoooka, Nishi-ku, Fukuoka-shi, Fukuoka 819-0395, Japan.*

<sup>4</sup>*Department of Systems Innovation, Graduate School of Engineering Science, Osaka University,  
Machikaneyama-chou, Toyonaka-shi, Osaka 560-8531, Japan*

*E-mail: {tokuo-tsuji, suzuki, twata, hseki}@se.kanazawa-u.ac.jp*

## Abstract

This paper presents a method for planning a robot motion of daily tasks by learning the relationship between object shapes and human motions. Robots are required to be able to deal with multifarious objects in various categories. However, it is difficult for robots to plan motions automatically for performing a task because objects even in the same category have different shapes. In our method, the motions are estimated by learning the relationship between object shapes and human motions using linear regression analysis.

**Keywords:** Motion Planning, Learning by demonstration, Machine Learning, Linear Regression Analysis, Robot hand

## 1. Introduction

Robots are expected to handle various objects in the field such as variable volume production, sorting of logistics, and daily tasks of life support. Since industrial objects and daily-life objects have different shapes even in the same category, robots have to adjust motions according to the difference in object shapes. For example, it is considered that the motion for pouring water with a bottle is generated by human teaching. Even though the motion can be used for the bottles which have the same shape, the motion may not be used for the bottles which have different shapes. The trajectory and execution time should be different according to object shapes. Since it is impossible to teach motions of every shapes manually,

the method for planning motions automatically according to object shapes is needed. Most motion planners of previous works are used for pick-and-place task and assembly task. However, it is difficult to plan a complicated motion like pouring water by using them.

We propose a method for planning motions by learning the relationship between shapes and human



Fig. 1. Overview.

motions. We applied the proposed method to the task of pouring water from plastic bottles (Fig.1).

Our main contributions in this paper are as follows.

- (1) Estimating shapes from measurement data using an RGB-D camera with one viewpoint
- (2) Learning the relationship between human motions and object shapes for the motion generation in advance
- (3) Acquisition object shapes and human motions by using an RGB-D camera and motion capture system
- (4) Planning a motion automatically according to shape parameters acquired automatically
- (5) Generating a motion for novel objects which has different shapes in the same category without a trial

Motions of pouring water from bottles into a cup are correlated with shapes of plastic bottles and cups. However, we assume the following things to verify the effectiveness of our proposed method. The object shapes except the manipulated object are the same. The initial water mass in the plastic bottles are known and constant.

## 2. Related works

In daily tasks, a task of folding cloth has been learned by robots based on working images [1]. Ochi et al. [2] proposed for scooping powder with a specific spoon by using deep learning. Roza et al. [3] performed the pouring motion by learning time series data of joint angles and force sensor from pouring motion by a robot which human manipulated. Schenck et al. [4] poured a specific amount of liquids for different containers by using real-time visual feedback in a PID controller and determining the amount of liquid in a container from vision sensor during a pouring action. Lopez et al. [5] optimize pouring motion for novel containers as a receiver efficiently by learning the motion to mitigate spillage from the simulation of the pour. Kroemer et al. [6] learned from the information of object shapes and human demonstrations and generalize the pouring motion for novel containers by using a trial-and-error approach quickly. Tamosiunaite et al. [7] learned motion primitives by reinforcement learning. The pouring motion was performed efficiently because the learned motion was adapted to novel objects and the motion was relearned. Although the shapes of containers as a receiver were considered, the shapes of pouring containers were not considered in these researches or they needed some trials to perform pouring motions.

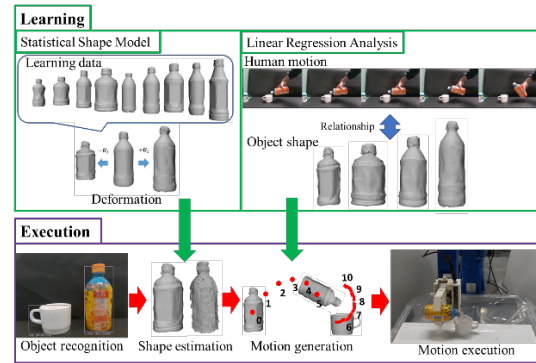


Fig. 2. Overview of motion generation.

It is necessary to handle the different shapes of objects so that motions can be planned with consideration for tasks, objects, and environments simultaneously. However, the previous researches use simple rotation for pouring motions and is not adapted to arbitrary trajectory which depends on the shape of bottles. In this study, we propose a motion trajectory generation to handle different shapes of manipulated objects.

## 3. Overview of motion generation

A flow chart of the overview is shown in Fig.2. The proposed method consists of two phases: learning and execution.

In the learning phase, two types of learning are performed. The first type constructs a statistical shape model (SSM) [8] from learning data collected from objects in the same category with different shapes. The second type derives relational equations between shapes and time series of human motions via linear regression analysis. The learning phase is completed prior to the execution phase.

In the execution phase, the shape of an object is estimated using an SSM. Motions are estimated by the linear regression equations based on the parameters of the shape obtained by the SSM.

## 4. Modeling and estimation of shapes by the SSM

SSMs of main objects are prepared for each category in the learning phase. In the execution phase, after recognizing the category of an object, its shape is estimated by using an RGB-D camera and expressed by various parameters. Shape estimation is performed by

using point clouds from depth images and contours from RGB images. An SSM describes an average shape and variations of multiple shapes that vary between different objects. A three-dimensional point cloud is resampled by mapping it onto an arbitrary target surface [8].

The shape vector  $X_i (i = 1, \dots, n)$  of an SSM is represented as follows:

$$X_i = [x_{i0}, y_{i0}, z_{i0}, \dots, x_{im}, y_{im}, z_{im}] \quad (1)$$

where the number of points is  $m$  and the number of shape models is  $n$ . A deformed shape vector  $X_i$  for the SSM is obtained by summing the calculated average shape vector  $\bar{X}$  and vectors  $U_i$  as follows:

$$x = \bar{X} + \sum_{i=1}^n U_i b_i \quad (2)$$

where the values of the shape parameters  $\beta = [b_1, b_2, \dots, b_n]$  represent the weight vector for each vector. The deformed shapes for each category are expressed by changing  $\beta$ .

Main objects are measured from one viewpoint by an RGB-D camera for shape estimation. A point cloud of the surface of a main object is extracted from a depth image. The contours of a main object are extracted from an RGB image. The iterative closest point algorithm (ICP) is used for registration between the point cloud of a depth image and that of an SSM. 50 SSMs were generated via random deformation for finding the parameters which have the minimum error. This procedure repeats until the shape parameters  $\beta$  are constant for ten iterations. Some examples of shape estimation via SSM are presented in Fig. 3.

## 5. Motion generation

The shape of a main object is represented by shape parameters  $\beta$ . The relationships between the shape parameters  $\beta$  and a human motion are learned and modeled by using linear regression analysis during the learning phase. The motion of a main object is estimated by using linear regression analysis during the execution phase.

### 5.1. Learning data acquisition

The speeds of human motions are different in each iteration. Therefore, we divide a motion into several smaller motions at various key frames and acquire learning data for an execution time  $t$ . The divided motions are defined as motion1, motion2, ..., and motion $N_t$ .

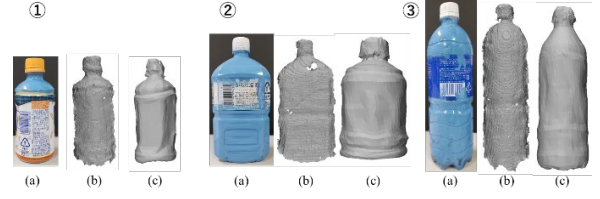


Fig. 3. Shape estimation (a) RGB Image, (b) Measurement point cloud, (c) SSM.

### 5.2. Linear regression

Linear regression is applied to the data for the shape parameters and motions and a regression coefficient matrix is calculated. Let  $n$  be the number of learning models for the construction of an SSM.

Let  $N$  be the number of learning data for linear regression. Only motion in a two-dimensional plane is considered. The position in the horizontal direction is defined as  $y$  and the position in the vertical direction is defined as  $z$ . The rotation angle in the  $yz$  plane is defined as  $\theta$ .

A linear equation is formulated from the shape parameters  $B = [\beta_1, \beta_2, \dots, \beta_N]^T$  and the position in the horizontal direction at a particular time  $Y = [y_1, y_2, \dots, y_N]^T$ . The normal equation for the linear equation  $Y = BP$  is solved as follows:

$$P = (B^T B)^{-1} B^T Y \quad (3)$$

A numerical solution  $(B^T B)^{-1}$  is derived via singular value decomposition. The posture and the position in the vertical direction at a particular time are also calculated. The regression coefficient matrices for the execution time are obtained in the same manner for motion1, motion2, ..., and motion $N_t$ . The products of the shape parameters  $\beta$  from novel objects and regression coefficient matrices  $P$  are calculated and the relative position and posture of the main object for each section (time series), as well as the execution time  $t$  for each motion are estimated.

## 6. Evaluation

We evaluated the proposed method for pouring water into a cup from a plastic bottle. The generated motions were evaluated experimentally using an actual robot. Shape models of nine plastic bottles were used for constructing an SSM with  $n=9$ . Ten real plastic bottles that were different from the nine shape models for the SSM were used in our experiments. Their point cloud data were

measured and shape parameters  $\beta$  were estimated 10 times for each bottle. Human motions of pouring water into the cup were performed 10 times for each bottle. The initial water mass in each bottle was 0.2 kg and 0.1 kg of water was poured into the cup. The number of the dataset is 100 sets in total.

The data from nine out of the ten plastic bottles were used as learning data ( $N=90$ ) and the data from the remaining bottle were used as testing data. Robot motions were generated by estimating the positions and postures of main objects for each section, as well as the execution times for each motion based on shape estimation of the testing data. We used five sequences per object and performed 50 experiments using an actual robot.

In these experiments, the motions generated based on the prepared dataset were verified by using an actual robot. The experimental conditions were as follows. The initial position and posture of the cup and plastic bottles were determined manually. The grasping points of the plastic bottles were also determined manually. The initial water mass in the plastic bottles was 0.2 kg. The experiment is shown in Fig. 4. The success rates of avoiding spilling water is 92% (46/50).

## 7. Conclusions

We proposed a method for generating motions by estimating the positions and postures of main objects and the execution time for each motion based on shape estimation. We performed position and posture estimations and obtained results very close to the variation of positions and postures in human motions. In experiments using an actual robot, the success rate for pouring water into a cup was 92% in terms of avoiding spilling water. In the future, we plan to generate motions based on shape changes in target objects, the initial water masses in plastic bottles for generalization of the proposed method.

## References

1. P.-C. Yang, K. Sasaki, K. Kase, S. Sugano and T. Ogata, Repetable Folding Task by Humanoid Robot Worker using Deep Learning, IEEE Robotics and Automation Letters(RA-L), pp. 397-403, 2017.
2. H. Ochi, W. Wan, Y. Yang, N. Yamanobe, J. Pan and K. Harada, Deep Learning Scooping Motion using Bilateral Teleoperations, Proc. of IEEE International Conference on Advanced Robotics and Mechatronics, 188, 2018.

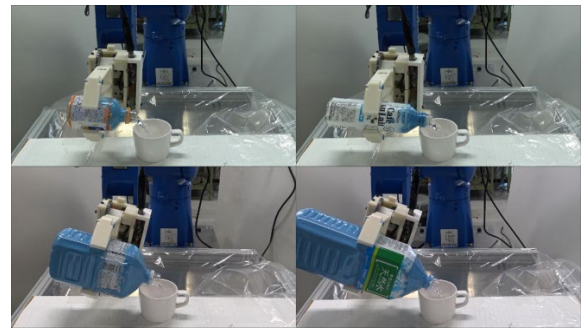


Fig. 4. Experiment.

3. L. Rozo, P. Jim'enez, and C. Torras, Force-based robot learning of pouring skills using parametric hidden Markov models, Robot Motion and Control, 2013 9th Workshop on. IEEE, 2013, pp. 227-232.
4. C. Schenck and D. Fox, Visual closed-loop control for pouring liquids., In 2017 IEEE International Conference on Robotics and Automation, pp. 2629-2636. IEEE, 2017.
5. T. Lopez-Guevara, N. K. Taylor, M. U. Gutmann, S. Ramamoorthy, and K. Subr, Adaptable Pouring, Teaching Robots Not to Spill using Fast but Approximate Fluid Simulation, Proceedings of the 1st Annual Conference on Robot Learning, vol.78, pp. 77-86, 13-15 Nov 2017.
6. O. Kroemer, E. Ugur, E. Oztop, and J. Peters, A kernel-based approach to direct action perception, in 2012 IEEE International Conference on Robotics and Automation, may 2012, pp. 2605-2610.
7. M. Tamosiunaite, B. Nemec, A. Ude, and F. Worgotter, Learning to pour with a robot arm combining goal and shape learning for dynamic movement primitives, Robotics and Autonomous Systems, vol. 59, no. 11, pp. 910-922, 2011.
8. S. Miyauchi, K. Morooka, Y. Miyagi, T. Fukuda, T. Tsuji and R. Kurazume, Tissue Surface Model Mapping onto Arbitrary Target Surface Based on Self-organizing Deformable Model, Workshop on Computer Vision at the 4th International Conference on Emerging Security Technologies, pp.79-82, 2013.