

Personalization Framework for Adaptive Robotic Feeding Assistance

Gerard Canal, Guillem Alenyà, and Carme Torras

Institut de Robòtica i Informàtica Industrial, CSIC-UPC
Llorens i Artigas 4-6, 08028 Barcelona, Spain
{gcanal, galenya, torras}@iri.upc.edu

Abstract. The deployment of robots at home must involve robots with pre-defined skills and the capability of personalizing their behavior by non-expert users. A framework to tackle this personalization is presented and applied to an automatic feeding task. The personalization involves the caregiver providing several examples of feeding using Learning-by-Demonstration, and a ProMP formalism to compute an overall trajectory and the variance along the path. Experiments show the validity of the approach in generating different feeding motions to adapt to user’s preferences, automatically extracting the relevant task parameters. The importance of the nature of the demonstrations is also assessed, and two training strategies are compared.

Keywords: Assistive Robotics, Personalized Human-Robot Interaction, Feeding, Trajectory Adaptation

1 Introduction

People with reduced mobility tend to find themselves needing the help of another in order to do the most basic tasks. Hence, performing Activities of the Daily Living (ADLs) such as eating, dressing, grooming or cleaning up can become very difficult. Intelligent robotic systems have proven useful in these situations by performing the helping task and, so, removing the constraint of constant attention from another person.

However, in order to effectively assist a human user the helping robot should be able to adapt to the specific user needs and preferences. Rather than performing a generic action suitable for anyone and forcing the user to adapt to the robot, it is the robot who should adapt its behavior taking into account the user and the situation, just as a human carer would do. The *empowering* of disabled people is crucial [3], and can be attained by providing more autonomy, intimacy and better quality of life. This does not imply the substitution of the caregiver, as personal contact is also very important. Contrarily, our approach relies on the caregiver to personalize the robot to the disabled person preferences.

In this paper, we first propose a novel Robot Personalization framework named FUTE (detailed in Section 3), that takes into account the user and allows concrete adaptation of generic pre-trained skills. In our framework, the robot is

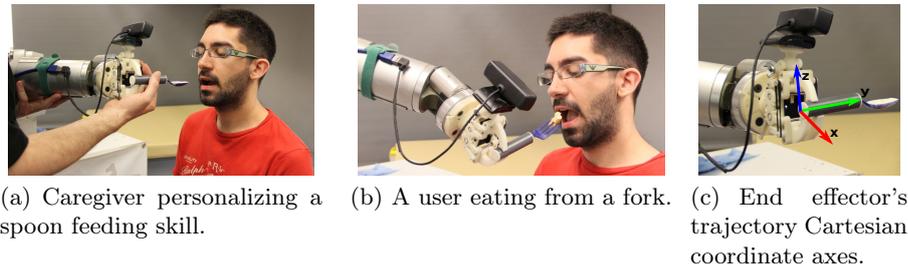


Fig. 1. Assistive personalized feeding application example.

pre-trained at the factory with a set of skills. Afterwards, when it arrives at the user's home, a non-expert teacher (the user itself or a caregiver) must have the freedom to adapt such skills to his/her preferences, or even teach new ones.

Second, we explore how to perform this training by using Learning-by-Demonstration techniques combined with a compliant robot control [7]. We propose two different interaction strategies: the teacher intervening in the robot motion, and the demonstration of a completely new trajectory.

In the third place, we test the applicability of the proposed FUTE framework in an assistive task consisting in feeding a person. As feeding can be very complex, we concentrate in a specific aspect: the way in which the robot approaches the cutlery to feed the person (see Fig. 1). We will show how our system is able to automatically extract the relevant aspects along the feeding task. Observe that, depending on the mobility and preferences of the user, the robot must wait with the food at some distance or introduce the food inside the mouth. Moreover, the feeding motion has to be adapted to the kind of food, for example yogurt or fries as seen in Fig. 1(b).

2 Related Work

Personalized Human-Robot Interaction has been studied in different works and fields. In education, it has been applied to Socially Assistive Robot (SAR) tutors that support the teaching task [5,10,12]. Baraka and Veloso [2] define three user models to adapt the luminous interactions between a robot and the user over time, learning the model parameters from user feedback. Personalized collaboration is shown in Fiore *et al.* [8], where an object manipulation task is performed jointly by the robot and the user whose preferences are taken into account. Abdo *et al.* [1] predict user preferences to tidy up objects in containers using collaborative filtering based on crowdsourced data and the observations of current dispositions or by querying the user. Although this strategy seems good for the tidying up task, it would not suit to capture the user preferences in an interaction context such as ours. Chernova and Veloso [4] present the Confidence-Based Autonomy (CBA) algorithm, which enables the agent to request demonstrations

from a human teacher, and allows him to correct further mistakes with additional demonstrations. The idea is similar to the User Tailoring one, though they apply it to improve the policy rather than to adapt a well-learned task to a specific user. A framework to learn and generalize complex tasks from unstructured demonstrations is proposed in Niekum *et al.* [14]. The method is able to recognize repeated instances of skills and generalize them to new settings.

In addition, more in the scope of this paper, personalized dressing assistance is performed by Gao *et al.* [9], where a user’s movement space is modelled and used to put on a sleeveless jacket. Similarly, Klee *et al.* [11] assist a user to place a hat in a collaborative way by means of asking the user to reposition itself when some user specific constraints do not hold. However, the personalization they propose consists in adapting to the user state or pose, but do not allow the user to modify the way in which the assistance will be carried out.

Moreover, we will apply the personalized interaction to the feeding scenario. Assistive feeding devices have been around for a while, mainly due to the evident need that some individuals have. Devices such as SECOM’s MySpoon [16,18] or the Handy 1 [17], among others, can provide significant help to allow people with upper limb disabilities to eat in a more autonomous manner.

Nonetheless, these systems lack the ability to adapt to the needs of each specific user. And, in cases of people with disabilities, this is a key factor for the system to be actually helpful in different kinds of environment, in which there is a handful of ways of assisting in the eating task, as often pointed out by long-term care nurses.

3 The FUTE Personalization Framework

We present a three-phase framework, the “FUTE framework”, to design and develop such kind of adaptive assistive applications. The three phases are called “Factory setting”, “User Tailoring” and “Execution tuning”, and are described as:

1. **Factory setting:** the robot is provided with the skills needed to perform the assistive task in a generic way. This would suit either the design of a new robot or the enhancement of an existing platform to carry out a new task.
2. **User Tailoring** (the focus of the paper): This second phase takes place in the user’s home. The robot performs a nominal skill, but personalization is encouraged in order to adapt its behavior to the user needs. In this phase, the robot should acquire, as automatically as possible, information about how the task has to be done for the user at hand while it performs the task in the generic way. This personalization may be done by the user or by an external agent (such as a carer), and it could be either explicit or implicit. In the feeding example, this will consist in the selection of the feeding point, it being either inside or outside the mouth. The data in our implementation include the proprioceptive robot perception as well as 3D images from a camera located at the hand of the robot (Fig. 1).
3. **Execution tuning:** In this last phase, the robot performs the task designed in the first phase but taking into account the personalization introduced in

Algorithm 1 Feeding execution

```

1: graspFeedingUtensil()                                ▷ Grasp a spoon or a fork.
2: repeat
3:   pickUpFoodFromPlate()
4:   userPose ← getHumanPoseFromPerception()
5:   moveToInitialPosition(userPose, initialPose)
6:   moveUtensilToFeedingPose(userPose, feedingPoint)    ▷ Approach the food
7:   waitForFoodConsumption()
8:   moveAwayFromUser(userPose)
9: until feedingIsComplete()                            ▷ User has had enough food or plate is empty

```

the second one. In the feeding example, the 6D pose of the user is computed using an RGBD camera and a face detection algorithm, and the robot trajectories are adapted to the current pose of the user. If the user is not satisfied with the robot behavior, the User Tailoring phase can be triggered again.

4 Experimental assessment: User-Centered Feeding Assistance

To build intuition, we will illustrate the different aspects of our proposal using the robot feeding application. Eating is one of the most basic physiological needs all human beings have, appearing at the base of Maslow’s hierarchy of needs [13]. However, some people with disabilities may not be able to do it by themselves, requiring the help of an external agent (usually a human carer), who will feed them taking into account their needs and capacities. To illustrate this, in the following experiments we tackle two example use-cases in which different personalizations can be applied: (U1) a person with very limited upper body mobility will require the caregiver to do all the feeding action, while (U2) another patient with upper limb disabilities may be able to move and eat the food by himself when it is close enough.

4.1 The Robot Feeding Process

Five steps can be identified for the adaptive feeding application (see Alg. 1). In the context of the proposed framework, steps between lines 1 and 3 would be provided to the robot during the factory training phase, while steps between lines 6 and 8 would be personalized at home. Thus, the complete execution is the outcome of joining the already known steps (at the factory phase) with the personalized ones, resulting in a successful feeding action for a specific person. The “**initialPose**” (line 5) and “**feedingPoint**” (feeding moment of the trajectory, line 6) parameters are obtained during the User Tailoring phase, as seen in Alg. 2. Note that in execution, the user can move freely. A vision system comprised of a low range RGBD sensor is used to compute their pose, and the robot motion is updated accordingly to obtain the desired feeding movement.

The vision system is also used to detect the moment in which the user bites the food in the “waitForFoodConsumption” step (line 7).

For the scope of this paper, we will just focus on the steps involving the user (lines 6, 7 and 8 from Alg. 1), and how they can be personalized to different users¹.

The feeding setup used in the experiments can be seen in Fig. 1(a) and Fig.1(b), and the coordinate axes at the robot’s end-effector are shown in Fig. 1(c). In it, the y Cartesian axis represents the frontal distance to the user, the x axis corresponds to the horizontal displacement and the z to the vertical one (the feeding height).

4.2 Feeding personalization

The User Tailoring strategy for feeding is shown in Alg. 2. It comprises the recording of N sample trajectories (line 4) including the approaching motion, waiting for the user to start the consumption, and a receding motion. The N trajectories are then used to learn a Probabilistic Movement Primitive (ProMP) [15,6] of the feeding movement (line 23). ProMPs are movement primitives that encode the time-varying variance of a set of trajectories. The state vector \mathbf{y}_t is defined as

$$\mathbf{y}_t = \begin{bmatrix} q_t \\ \dot{q}_t \end{bmatrix} = \mathbf{\Phi}_t^T \mathbf{w} + \epsilon_y, \quad (1)$$

where $\mathbf{\Phi}_t = [\phi_t, \dot{\phi}_t]$ is the time-dependent basis matrix, \mathbf{w} is the weight vector and $\epsilon_y \sim N(0, \Sigma_y)$ is Gaussian noise. The trajectories can then be represented as a mean trajectory and its variance, each time point being represented as $\mu_t \pm \sigma_t$. New trajectories can be sampled from the distribution, and via points are defined using the conditioning operator. We have used the ProMP formalism because, apart from the trajectory itself, as will be seen in Section 4.4, it also provides insights of the particularities of the task by means of the variance along the trajectory.

We would like to assess the impact of variations in the demonstrated trajectories, to provide hints to the caregiver demonstrating the task about how similar the N demonstrations should be. The next experiment tackles use-case U1: introducing the food inside the mouth of the user. It involves demonstrations using two different feeding paths with a mannequin as user: the first one in which the carer tried to perform the same trajectory 5 times, and the second set in which the 5 trajectories had different approaching movements (but with the same feeding point). The results are shown in Fig. 2.

Comparing Figs. 2(a) and 2(b) it can be seen that the shape of both mean trajectories is quite alike, both reaching the same feeding position (shaded area). As a consequence, apparently there is no need to have several similar trajectories in order to have a good average feeding movement. However, we observe different variances. In Fig 2(a) variance is almost constant during the whole trajectory,

¹ A video showing the process of the personalized feeding task can be found at www.iri.upc.edu/groups/perception/frameworkFUTE.

Algorithm 2 User tailoring strategy

```

1: demonstrations  $\leftarrow \emptyset$  ▷ Will store the new recorded trajectories
2: feedingPoints  $\leftarrow \emptyset$  ▷ Time points of each trajectory in which person was fed
3: initialPoses  $\leftarrow \emptyset$  ▷ Face pose at the start of each trajectory
4: for N do
5:   if unassistedTraining then ▷ Set one of the two personalization modes
6:     SetRobot(gravityCompensationMode)
7:   else
8:     SetRobot(ReproduceFactoryTrajectory, stiffness)
9:   end if
10:  initialPoses  $\leftarrow$  append(getUserFacialPose())
11:  newTrajectory  $\leftarrow \emptyset$ 
12:  while robotMoving do ▷ Store approaching trajectory
13:    addPoints(newTrajectory)
14:  end while
15:  waitForFoodConsumption() ▷ Wait until user starts eating
16:  feedingPoints  $\leftarrow$  append(currentTrajectoryPoint)
17:  while robotMoving do ▷ Store receding trajectory
18:    addPoints(newTrajectory)
19:  end while
20:  demonstrations  $\leftarrow$  append(newTrajectory)
21: end for
22: referenceFeedingPoint  $\leftarrow$  alignToFeedingPoint(demonstrations, feedingPoints)
23: personalizedTrajectory  $\leftarrow$  RecomputeProMP(demonstrations)
24: return <personalizedTrajectory, referenceFeedingPoint, avg(initialPoses)>

```

while in Fig. 2(b) variances in the approaching and receding movements are larger, but smaller in the feeding point. Observe that obtaining this information is crucial, as the robot should act carefully while feeding the user (lower variance) whereas approaching and receding can exhibit a more careless behavior (larger variance). Thus, we conclude that showing some variability in the demonstrated trajectories is important.

4.3 Teaching Modes: Unassisted vs. Compliant Reproduction

Two teaching modes have been defined (Alg. 2 lines 5–9). The first one is unassisted, the robot only compensates gravity and the caregiver has to start from scratch each demonstration handling the robot and freely performing a feeding trajectory. This allows the user to discard the factory settings and re-teach the whole movement. In the second one, the robot executes a generic feeding trajectory –which was recorded in the factory setting phase– using a compliant controller [7] that uses a *stiffness factor* to determine the arm’s stiffness degree.

The next experiment is designed to assess the effect of the stiffness factor. Hence, we repeated the executions with different stiffness values for the same trajectory where the caregiver personalized the motion so that the feeding occurred further away from the person (a mannequin was used in this experiment

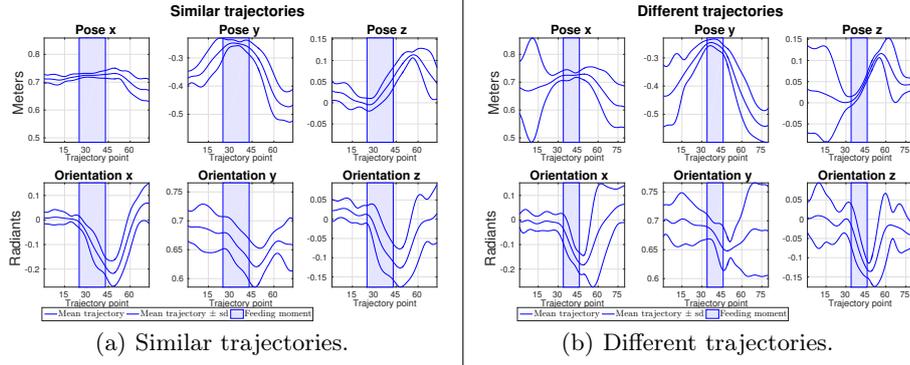


Fig. 2. Comparison between similar and different example trajectories. The thicker line is the mean trajectory, and the surrounding lines are the mean \pm standard deviation. The shaded regions denote the part of the trajectory in which the food is consumed.

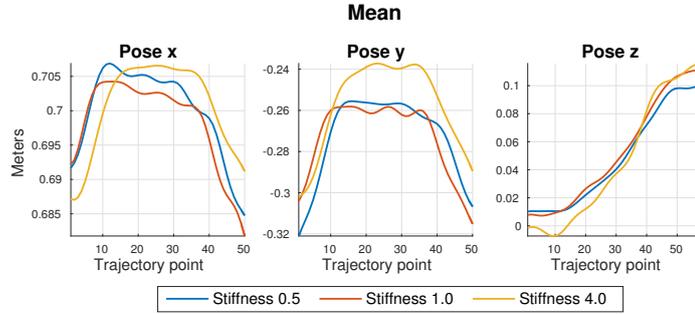


Fig. 3. Mean trajectories generated from a default trajectory with different stiffness values. The lower the stiffness, the most docile the robot behaviour is. Observe the oscillations introduced when the trajectory is perturbed.

to avoid noise induced by involuntary movements and ease the comparison). Here we tackle use-case U2: the trajectory is modified to end outside of the mouth, for instance for patients some mobility. The results are shown in Fig. 3.

The intuition says that starting from scratch at every demonstration is harder, whereas if the robot reproduces the movement in a docile manner the user only has to physically perturb the execution in some parts and teaching becomes easy. However, as it can be seen in Fig. 3, this second approach introduces oscillations of about half a centimeter in the resulting trajectory, not only in the y axis (the approaching direction) but also in x and z . With low stiffness values the oscillations tend to be higher as the robot reacts to slighter perturbations as when it tries to go on with the trajectory and return to the original path and the user holds it again. In contrast, higher stiffness makes it harder for the user to modify the trajectory, resulting in less oscillations but more physical effort for the user.

4.4 Automatic parameter extraction

In the next experiment, the modifications that the caregiver can introduce to personalize the feeding process are (see Alg. 2, lines 10, 20 and 22): the initial pose, the motion shape, and the feeding point (inside the mouth for use-case U1 or just approaching the food for use-case U2).

We show how these parameters can be extracted automatically during the User Tailoring phase. First, the feeding point is computed by recording the distance to the face in which the movement is stopped to feed the person. Second, the motion learning process captures the particularities of the task. We exemplify this fact by observing variances of the ProMP trajectory related to two different utensils: when a spoon is used the orientation is more restricted, while a fork allows for more flexibility.

In this experiment, the re-teaching has been carried out with the robot holding a spoon with yogurt and also with a fork pinching a french fry. Five trajectories were recorded in order to generate the ProMP for each case. A human user was used here as test subject (not a mannequin) because the insertion orientation was relevant (see Fig. 1). With this experiment, we can observe how the particularities of the task are integrated into the ProMP. Figs. 4(a) and 4(b) show the trained trajectories for each Cartesian coordinate and the rotations around each axis, displaying the mean trajectory and its variance.

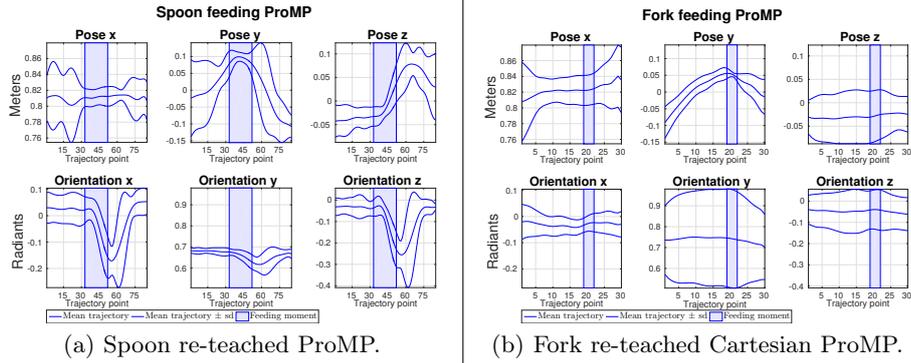


Fig. 4. Learned trajectories for the spoon and fork experiments, where the gravity compensation mode was used for re-teaching the trajectories.

The figures clearly show the moment in which the utensil is near the mouth (as seen in the shaded regions), because the variance of the movement narrows at that stage. This is, in fact, a representation of the flexibility of the movement, since the critical parts that need more precision are less flexible.

In addition, this variance effect can also be seen in the orientation plots, in which the spoon’s sample orientation variances are narrower at the beginning of the trajectory to avoid spilling the content, while the move away part has

wider variances as the food has already been taken. The fork trajectory has less restrictive orientations because there is less danger of dropping food, as clearly seen in the orientation around the y axis.

Moreover, this gives us insights on how the variance in the trajectory points provided by the ProMP could also be used to control the compliance (stiffness degree) of the robot during the trajectory execution phase. This way, the robot would be more docile to external forces in moments of high variance, corresponding to points of the path that have been taught in non precise ways, and more rigid in low variance points. Thus, the robot would not react to external forces while introducing the spoon in the mouth, avoiding any possible harm to the user due to accidental robot perturbations. Note this should not be applied in the joints interacting with the user, allowing for docile movement with the mouth but being stiff in external joints such as the elbow.

5 Conclusions

In this paper, we presented the FUTE robot personalization framework consisting of three phases: Factory setting, User Tailoring and Execution tuning. This framework has been devised to help the implementation of assistive applications by allowing easy adaptation of the assistive robot performance to specific users, given the fact that all of them are different and have their own special needs. Furthermore, it allows non-expert users to conduct the robot adaptation just by guiding the robot behavior.

Then, we tested this framework in a feeding application where a human caregiver can re-teach the feeding movement the robot has to perform, by physically modifying an already learnt trajectory or by teaching it from scratch. This allows the person to teach the feeding point and distance so it can be either inside or near the mouth. Moreover, we demonstrate how the use of the Probabilistic Movement Primitives (ProMPs) is an appropriate choice for these kind of assistive applications, as they are able to learn the particularities of the task, such as the feeding moment and the flexibility of each part of the trajectory.

Nevertheless, the feeding application has still room for improvement, including, but no limited to, the integration of voice commands with the physical interactions, the adaptation of the best stiffness factor for the carer that is performing the re-teaching or the use of the ProMP trajectory variance to control the stiffness factor while feeding the user.

Acknowledgments. This work has been supported by the MINECO project RobInstruct TIN2014-58178-R and the ERA-Net CHIST-ERA project I-DRESS PCIN-2015-147. Gerard Canal is also supported by the Ministry of Economy and Knowledge of the Government of Catalonia via a FI-DGR 2016 fellowship.

References

1. Abdo, N., Stachniss, C., Spinello, L., Burgard, W.: Robot, organize my shelves! Tidying up objects by predicting user preferences. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 1557–1564 (2015)
2. Baraka, K., Veloso, M.: Adaptive interaction of persistent robots to user temporal preferences. In: International Conference Social Robotics (ICSR). pp. 61–71 (2015)
3. Chen, T.L., Ciocarlie, M., Cousins, S., Grice, P.M., Hawkins, K., Hsiao, K., Kemp, C.C., King, C.H., Lazewatsky, D.A., Leeper, A.E., Nguyen, H., Paepcke, A., Pantofaru, C., Smart, W.D., Takayama, L.: Robots for humanity: using assistive robotics to empower people with disabilities. *IEEE Robotics Automation Magazine* 20(1), 30–39 (2013)
4. Chernova, S., Veloso, M.: Interactive policy learning through confidence-based autonomy. *Journal of Artificial Intelligence Research* 34, 1–25 (2009)
5. Clabaugh, C., Ragusa, G., Sha, F., Matarić, M.: Designing a socially assistive robot for personalized number concepts learning in preschool children. In: Int. Conf. on Development and Learning and Epigenetic Robotics. pp. 314–319 (2015)
6. Colomé, A., Neumann, G., Peters, J., Torras, C.: Dimensionality reduction for probabilistic movement primitives. In: IEEE-RAS International Conference on Humanoid Robots. pp. 794–800 (2014)
7. Colomé, A., Pardo, D., Alenyà, G., Torras, C.: External force estimation during compliant robot manipulation. In: IEEE International Conference on Robotics and Automation (ICRA). pp. 3535–3540 (2013)
8. Fiore, M., Clodic, A., Alami, R.: On planning and task achievement modalities for human-robot collaboration (2016)
9. Gao, Y., Chang, H.J., Demiris, Y.: User modelling for personalised dressing assistance by humanoid robots. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 1840–1845 (2015)
10. Greczek, J., Short, E., Clabaugh, C., Swift-Spong, K., Matarić, M.J.: Socially assistive robotics for personalized education for children. In: AAAI Fall Symposium on Artificial Intelligence and Human-Robot Interaction (2014)
11. Klee, S.D., Ferreira, B.Q., Silva, R., Costeira, J.P., Melo, F.S., Veloso, M.: Personalized assistance for dressing users. In: International Conference Social Robotics (ICSR). pp. 359–369 (2015)
12. Leyzberg, D., Spaulding, S., Scassellati, B.: Personalizing robot tutors to individuals’ learning differences. In: ACM/IEEE International Conference on Human-Robot Interaction (HRI). pp. 423–430. ACM (2014)
13. Maslow, A.H.: A theory of human motivation. *Psych. Review* 50(4), 370–396 (1943)
14. Niekum, S., Osentoski, S., Konidaris, G., Barto, A.G.: Learning and generalization of complex tasks from unstructured demonstrations. In: International Conference on Intelligent Robots and Systems (IROS). pp. 5239–5246. IEEE (2012)
15. Paraschos, A., Daniel, C., Peters, J., Neumann, G.: Probabilistic movement primitives. In: Advances in Neural Information Processing Systems (NIPS) (2013)
16. Song, W.K., Song, W.J., Kim, Y., Kim, J.: Usability test of KNRC self-feeding robot. In: International Conference on Rehabilitation Robotics. pp. 1–5 (2013)
17. Topping, M.: An Overview of the Development of Handy 1, a Rehabilitation Robot to Assist the Severely Disabled. *Intelligent and Robotic Systems* 34(3), 253–263 (2002)
18. Zhang, X., Wang, X., Wang, B., Sugi, T., Nakamura, M.: Real-time control strategy for EMG-drive meal assistance robot - My Spoon. In: International Conference on Control, Automation and Systems (ICCAS). pp. 800–803 (2008)