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

Pattern Analysis and Parameters Optimization of Dynamic Movement Primitives for Learning Unknown Trajectories

Mantian Li[1], Zeguo Yang[1], Fusheng Zha[1†,2], Xin Wang[2], Pengfei Wang[1],
Wei Guo[1], Darwin Caldwell[3], Fei Chen[3†], *Member, IEEE*

***Abstract*— A robot in the future may initially has a good learning capability but an empty library of movements. It gradually enriches its library of movements through human demonstrations. Dynamic Movement Primitives (DMPs) has been proved to be an effective way to represent trajectories. Trajectories are classified into discrete and rhythmic ones, and parameters are set for each demonstrated trajectory. However, what kind of trajectory will be provided by robot users is sometimes unknown to robot developers, so trajectory pattern and the parameters can not be determined in advance. It's also impossible for non-technical robot users to set these parameters and determine the pattern of movements they are going to demonstrate. To make it easier for non-expert robot users to programme their robots by demonstration, this work presents an efficient way to deal with these two problems. The effectiveness of the proposed methodology is proved by teaching a robot to clean the whiteboard in different ways and stack a set of cubic boxes in specific order.**

I. INTRODUCTION

Intelligent Robots will gradually become an important part of our work and daily life, making our life easier by helping us doing all kinds of work, such as cleaning and washing ,taking care of elderly and children, cooperating with human, etc. It may become common for a human to have several robotic assistants. One of the main challenges for robots to autonomously completing these tasks is the capability of generating required trajectories. As requirements on trajectories vary with tasks and situations, it is impossible to pre-programme robots for future tasks via traditional motion planning algorithms. Imitation learning provides a promising approach to this problem [1]. Within imitation learning, a control policy is learned from human demonstrations [2]. A robot user can teach a robot how to complete a task under certain situations via demonstrations, which even a

*This work was supported by Natural Science Foundation of China (Grant No.61773139), The Foundation for Innovative Research Groups of the National Natural Science Foundation of China (Grant No.51521003), Shenzhen Science and Technology Program (Grant No.KQTD2016112515134654) and Shenzhen Science and Technology Research and Development Foundation (Grant No.JCYJ20190813171009236).

†Corresponding authors: Fusheng Zha, Fei Chen.

[1]Mantian Li, Zeguo Yang, Fusheng Zha, Pengfei Wang and Wei Guo are with State key Laboratory of Robotics and Systems, Harbin Institute of Technology, 150001, Harbin, China (e-mail: `zhafusheng@hit.edu.cn`).

[2]Fusheng Zha (visiting scholar) and Xin Wang are with Shenzhen Academy of Aerospace Technology, 518057, Shenzhen, China (e-mail: `xin.wang@chinasaat.com`).

[3]Darwin Caldwell and Fei Chen are with Active Perception and Robot Interactive Learning Laboratory, Department of Advanced Robotics, Istituto Italiano di Tecnologia, Via Morego 30, 16163, Genova, Italy (e-mail: `fei.chen@iit.it`).

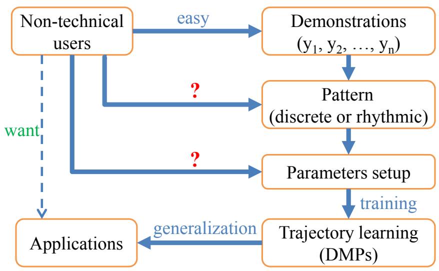


Fig. 1: Learning by demonstration with DMPs. Demonstration is easy, but it is hard for non-technical users to determine the pattern and set the required parameters

non-technical user is competent. The robot will learn skills from datasets of demonstrations and gradually create its own skill library.

How to simply and efficiently represent trajectories is one of the key issues of imitation learning. DMPs has been proved to be an effective representation of trajectories for several advantages, including high robustness and adaptation to new situations, temporal invariance, etc [3][4]. Both discrete and rhythmic motions can be represented by DMPs and generalized to new situations [5][6]. The method has been providing creative solutions for trajectory learning in a lot of directions, ranging from mobile manipulation [7] to exoskeleton robot control [8].

Moreover, in order to autonomously complete similar tasks in new situations with previously learned skills, a robot has to create and enlarge its skill repository through demonstrations provided by human or other robots [1]. S. Niekum et al present a DMPs-based framework to learn a large library of skills and multi-step tasks with PR2 mobile manipulator [9]. In [10], researchers build a library of movements by labeling each recorded movement and teach a robot to perform pick-and-place operations and water-serving tasks. G. Maeda et al introduce an active incremental learning algorithm so that a real robot arm would ask for extra demonstrations if skills in its repertoire are not enough to complete the commanded task [11]. F. Meier et al assume the existence of a library of movement primitives and provide an algorithm to segment and recognize complex human movement and enlarge the library when necessary [12].

However, only discrete movement primitives have been taken into account in the aforementioned research. Since rhythmic movements exist everywhere in our daily lives (such as writing [6], wiping [13] and drumming [14]), an intact library of motions should include both discrete and rhythmic movement primitives. According to the research presented in [5][6], discrete and rhythmic movement primitives should be learned with different models and be generalized to new situations in different ways. It raises a problem how to determine whether a movement primitive is discrete or rhythmic. In [15], researchers manually classify motions into discrete and periodic ones and match them with discrete and rhythmic models, respectively. In addition, some parameters exist in learning models of DMPs vary with tasks and must be properly set for each trajectory by hand. Nevertheless, what kind of trajectories will be demonstrated in the future by robot users is unknown. Pattern determination and parameters setup can not be completed in advance. As shown in Fig. 1, it is difficult and even impossible for non-technical users to determine the pattern of demonstrations and set the required parameters of learning models. To make it easier for non-technical robot users to perform demonstrations to their robots, our research presented in this paper will try to deal with these two issues. Characteristics of trajectories are numerically analyzed, so the pattern of demonstrations are determined without known the geometric information of trajectories. For both discrete and rhythmic movement primitives, parameters of learning models are obtained via Bayesian optimization (BO) [16] instead of manual tuning.

The contribution of this paper lies in two aspects. Firstly, we present a methodology to determine what kind of pattern of movement primitives should be applied to match an unknown trajectory provided by robot users. Furthermore, we provide an algorithm to obtain optimal parameters of both discrete and rhythmic dynamic movement primitives. These two contributions together finally build a bridge from unknown demonstrations to determinant representations of movement primitives. That is, the only thing left for a robot user to do is to provide demonstrations.

II. METHODOLOGY

We try to explain our methodology of analyzing the pattern of an unknown trajectory in this section. The principle of the proposed methodology can be briefly explained as follows. The premise for a trajectory to be represented by rhythmic movement primitives is that the duration of the demonstration must more than one period, otherwise, its period can not be determined. Ideally, for every point in the demonstration, there exists at least one other points that have the same value. If the trajectory satisfies the requirement, it will be temporarily treated as a periodic motion for further consideration. Otherwise, it should be considered as a discrete motion. Generally, for a periodic movement, all the maximum values should be almost equivalent, so should the minimum values. Furthermore, since the chosen model will be used to generalize to new situations, the effectiveness must be guaranteed. To verify the effectiveness of the chosen model, it will be virtually generalized to new situations, such as new goal positions (discrete model) or new amplitudes (rhythmic model). DMPs can also be utilized to recognize and classify movements [4]. If the chosen model matches the real attributes of the demonstrated trajectory, the generalized

Algorithm 1 Pattern analysis and parameters optimization of DMPs for known trajectories

Input: Demonstration dataset $D = \{y_1, \dots, y_n\}$
Output: $\alpha_z, nfs, Pattern$

$sample\ a\ set\ of\ m\ points\ SP$
$c \leftarrow 0$
for $y_i \in SP\ AND\ y_j \in D$ **do**
  if $y_i == y_j\ AND\ i \neq j$ **then**
    $c++$
  end if
end for
if $c/m > threshold_1\ AND\ EXTREMUMANALYSIS(D)$ **then**
  $Pattern \leftarrow rhythmic$
else
  $Pattern \leftarrow discrete$
end if
$DMPs(\alpha_z, nfs, \mathbf{w}_a) \leftarrow BO(DMPs(\alpha_z, nfs))$
\\ Obtain optimal parameters through BO
if $Pattern == discrete$ **then**
  $DMPs(\alpha_z, nfs, \mathbf{w}_a) \leftarrow NewGoal(NG)$
  $NewTrajectory(NT) \leftarrow DMPs(\alpha_z, nfs, NG)$
else
  $DMPs(\alpha_z, nfs, \mathbf{w}_a) \leftarrow NewAmplitude(NA)$
  $NewTrajectory(NT) \leftarrow DMPs(\alpha_z, nfs, NA)$
end if
$\mathbf{w}_b \leftarrow DMPs(\alpha_z, nfs, NT)$
$r_{ab} \leftarrow recogntion(\mathbf{w}_a, \mathbf{w}_b)$
if $r_{ab} > threshold_4$ **then**
  $result \leftarrow true$ \\ Pattern judgement is verified
else
  $\alpha_z^i \leftarrow 0, nfs^i \leftarrow 0, result \leftarrow false$
end if
function EXTREMUMANALYSIS(D)
  $MaximumSet(MA) \leftarrow D, MinimumSet(MI) \leftarrow D$
  \\ Calculate sets of minimums and maximums
  if $max(MA) - min(MA) < threshold_2\ AND\ max(MI) - min(MI) < threshold_3$ **then**
    return $true$
  else
    return $false$
  end if
end function

trajectory should be recognized by the chosen model. Then, the chosen model and corresponding parameters will be stored in the library of motions to represent the demonstrated movement. As long as the model to match the trajectory is determined, parameters of the corresponding model should be properly determined. Then, optimal parameters are obtained through Bayesian optimization. The methodology is displayed in **Algorithm 1** and will be given further explanation in the following subsections. For multi-dimensional demonstrations, the algorithm will be performed in each dimension.

A. Dynamic Movement Primitives

The methodology of representing trajectories with DMPs was proposed by Ijspeert et al [3][5] and will be briefly reviewed in this paper. Motions are classified into discrete and periodic ones. Furthermore, point attractor systems and limit circle attractor systems are applied to derive control policies for discrete and rhythmic movement primitives, respectively [5][6].

For a one dimensional trajectory $y(t)$, the desired attractor behavior can be achieved by the following formulations, which are also called the transformation system [4]

$$\tau \dot{v} = \alpha_z(\beta_z(g-y)-v)+f, \tag{1}$$

$$\tau \dot{y} = v, \tag{2}$$

where α_z and β_z are positive constants determined by developers and τ is a timescaling parameter. f is a nonlinear forcing term which determines the properties of a trajectory to a very large extent.

For discrete movement primitives, the term g in Eq. 1 denote the target position. The forcing term f can be chosen as

$$f(s) = \frac{\sum_{i=1}^{nfs} \psi_i(s)\omega_i}{\sum_{i=1}^{nfs} \psi_i(s)} s(g-y_0), \tag{3}$$

where y_0 is the start position, nfs denotes the number of basis functions and $\psi_i(s)$ are Gaussian functions expressed by

$$\psi_i(s) = exp(-h_i(s-c_i)^2), \tag{4}$$

where the terms h_i and c_i denote the width and center of the i_{th} Gaussian function, respectively. The variable s is a phase term to make the dynamical system temporal invariant and is driven by the canonical system [4]

$$\tau \dot{s} = -\alpha_s s, \tag{5}$$

where α_s is a positive constant.

For rhythmic movement primitives, the term g in Eq. 1 can be interpreted as an anchor point of a periodic trajectory. The forcing term f will be replaced by

$$f(\theta, r) = \frac{\sum_{i=1}^{nfs} \phi_i(\theta)\omega_i}{\sum_{i=1}^{nfs} \phi_i(\theta)} r, \tag{6}$$

where the term r is an amplitude signal which could be adjusted online, and Gaussian functions are formulated as

$$\phi_i(\theta) = exp(h_i(\cos(\theta - c_i)-1)), \tag{7}$$

where the term $\theta \in [0, 2\pi]$ denotes the phase angle of the movement primitive, and the following canonical system is utilized to replace Eq. 5

$$\tau \dot{\theta} = 1, \tag{8}$$

In the above analysis, weighting factors ω_i in both Eq. 3 and Eq. 6 can be learned via a locally weighted regression (LWR) algorithm [17]. According to [4], the terms c_i and h_i in Eq. 4 and Eq. 7 can be determined by the requirement that the Gaussian functions should be evenly distributed with respect to time, and the constants α_s and β_z are related to α_z by $\beta_z = \alpha_z/4$ and $\alpha_s = \alpha_z/3$. As a result, the constant α_z and the number of Gaussian functions nfs are left to be determined, which vary with trajectories and will affect the learning results. As has been mentioned above, it is inconvenient and even impossible for non-technical robot users to determine the appropriate values of these parameters. In this paper, we try to obtain optimal values of these parameters automatically.

B. Parameters Optimization

Bayesian optimization [16] has been proved to be an efficient optimization approach and has been widely utilized in machine learning [18] and robotics [19], so it is chosen to obtain optimal parameters in our research.

Usually, positions of several dimensions will be recorded in a real demonstration process, and each dimension will be matched with a transformation system. Without losing generality, we choose an one-dimensional trajectory as an example. The demonstration trajectory is denoted by $y_{demo}(t)$. Velocity $\dot{y}_{demo}(t)$ and acceleration $\ddot{y}_{demo}(t)$ are obtained by taking derivatives with respect to time. The goal of learning is set to definitely repeat the demonstrated motion. The trajectory learned by DMPs is represented by $y_{dmp}(t)$, $\dot{y}_{dmp}(t)$ and $\ddot{y}_{dmp}(t)$. The correct pattern with optimal parameters should repeat the goal trajectory with the least deviation, so the cost function is defined as

$$\varepsilon = k_1\varepsilon_p + k_2\varepsilon_v + k_3\varepsilon_a, \tag{9}$$

where k_i $(i = 1, 2, 3)$ are positive constants,

$$\varepsilon_p = \sum_{t=t_{start}}^{t=t_{end}} (|y_{demo}(t) - y_{dmp}(t)|), \tag{10}$$

$$\varepsilon_v = \sum_{t=t_{start}}^{t=t_{end}} (|\dot{y}_{demo}(t) - \dot{y}_{dmp}(t)|), \tag{11}$$

$$\varepsilon_a = \sum_{t=t_{start}}^{t=t_{end}} (|\ddot{y}_{demo}(t) - \ddot{y}_{dmp}(t)|), \tag{12}$$

denote the learning deviations caused by position, velocity and acceleration, respectively. The values of constants k_i $(i = 1, 2, 3)$ will be further discussed in section IV. After the Bayesian optimization process finishes, the least value of ε and corresponding optimal parameters (α_z and nfs) and weighting factors ($\mathbf{w}$) will be obtained, that is, the DMPs representation of the demonstration is determined.

C. Pattern Analysis

Discrete movement primitives are used for trajectory learning of point-to-point reaching movements (such as a pointing motion), while rhythmic movement primitives are used for that of periodic tasks (such as drumming tasks). These two kind motions should be clearly distinguished and fitted by different models. So it is of great importance to analyze whether a trajectory should be considered as discrete or rhythmic.

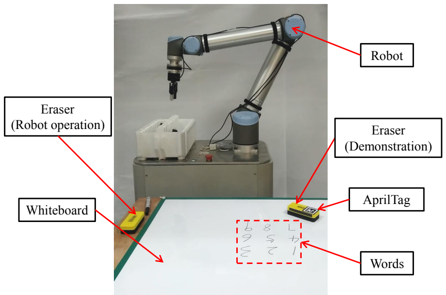


Fig. 2: Task of cleaning the whiteboard

As has been mentioned above, two requirements must be satisfied for a demonstration to be considered as a periodic movement: (1) all the point values should appear more than once; (2) all the maximum values and minimum values should be equivalent or have similar values, respectively. Moreover, according to the research on DMPs presented in [4], topologically similar trajectories tend to be fit by similar weighting vector $\mathbf{w}$, so the set of parameters can be applied to recognize similar trajectories. The recognition criterion is calculated by

$$r_{ab} = \frac{\mathbf{w}_a^T \mathbf{w}_b}{|\mathbf{w}_a||\mathbf{w}_b|}, \tag{13}$$

where $\mathbf{w}_a$ and $\mathbf{w}_b$ denote vectors formed by the weight parameters of the two trajectories.

A learned motion will be generalized to new situations in future applications. To verify the generalization capability of the representation in advance, it is "virtually" generalized to several new situations. The word "virtually" here means that we just obtain generalized trajectories through numerical calculation and will not perform the trajectories with real robots. Then, the generalized trajectories will be fit to the candidate model. We assume the vectors formed by weighting factors of the original and a generalized trajectory are denoted by $\mathbf{w}_o$ and $\mathbf{w}_g$, respectively. The recognition result r_{og} is calculated with Eq. 13 using $\mathbf{w}_o$ and $\mathbf{w}_g$. If all the errors between r_{og} and 1 is smaller than a prescribed value, the candidate model is definitely considered as the correct model. Furthermore, the corresponding learning model with optimal parameters will be applied to represent the demonstrated motion for future applications. Conversely, the generalization ability of the representation is not guaranteed and should be given further consideration.

III. EXPERIMENTS

Two tasks are chosen as examples to illustrate the efficacy of the proposed method: clean the whiteboard and stack boxes. All the weighting constrants k_i $(i = 1,2,3)$ in Eq. 9 are set to 1 in the two experiments. In order to achieve better learning result, the original trajectory is smoothed via a second order Butterworth filter [20]. Demonstration data is recorded by an Intel RealSense D435 depth camera at a frequency of 30Hz and is transformed into 1000Hz via spline interpolation. Limits of parameters for Bayesian optimization are set as $\alpha_z \in [0.01, 50], nfs \in [2, 200]$.

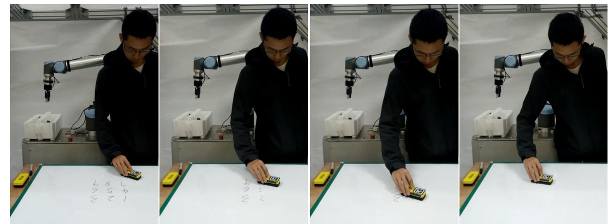


Fig. 3: Demonstration of cleaning the whiteboard

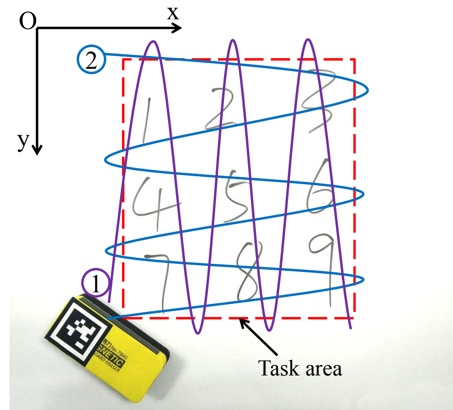


Fig. 4: Ways to wipe words off the whiteboard

A. *Clean the Whiteboard*

A six-dimensional robot is shown how to wipe the words off the whiteboard (Fig. 2). An AprilTag visual fiducial [21] is attached to the eraser used for demonstration, and the depth camera used to record the position of the eraser is mounted just on the top of the whiteboard. Since the orientation and the position along the upright direction of the eraser almost keep constant during the task, only the position information along the x-axis and y-axis will be used for trajectory learning. Another eraser is used for robot operation. Several demonstrations are performed and screenshots of one of the demonstration processes are shown in Fig. 3. As shown in Fig. 4, demonstrations can be performed in two modes, of which the properties are shown in Table I. Different demonstration modes would lead to different pattern of movements in the same directions. The area of words for demonstration is different from that for real robot operations, so the learned skills must be generalized to new situations.

In this experiment, 5 demonstrations are performed in each mode. For the purpose of analysis, starting points of demonstrations with the same method are transform to the same position, and the same number of sample points are extracted from original demonstrations. Table II displays the pattern analysis results and optimal values of corresponding parameters in the form "Pattern(α_z,nfs)" ("D" and "M" are short for "Demonstration" and "Mode", respectively).

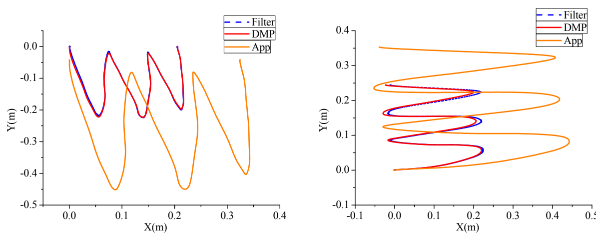
TABLE I: Properties of different demonstration modes

Mode / Dimension	Mode_1	Mode_2
x	Discrete	Rhythmic
y	Rhythmic	Discrete

TABLE II: Pattern analysis results and optimal parameters - Clean the Whiteboard

D \ M		$Mode_1$	$Mode_2$
1	x	Discrete (3.17, 138)	Rhythmic (0.57, 199)
	y	Rhythmic (23.98, 196)	Discrete (4.28, 199)
2	x	Discrete (12.38, 200)	Rhythmic (0.01, 199)
	y	Rhythmic (10.15, 178)	Discrete (19.86, 157)
3	x	Discrete (0.01, 200)	Rhythmic (2.63, 199)
	y	Rhythmic (9.63, 196)	Discrete (6.05, 183)
4	x	Discrete (14.71, 200)	Rhythmic (0.02, 199)
	y	Rhythmic (5.08, 179)	Discrete (3.22, 188)
5	x	Discrete (14.84, 199)	Rhythmic (12.11, 200)
	y	Rhythmic (0.01, 200)	Discrete (18.32, 199)

The proposed method has successfully learn correct representations with proper parameters of all demonstrations. Take the first demonstration of each mode as examples. The learned behaviors and generalized movements of the robot in real situations are displayed in Fig. 5 and Fig. 6 ("App" stands for information in real applications of experiments).



(a) (b)

Fig. 6: Resultant 2-dimensional trajectories of wiping demonstrations in different modes. (a) denotes learning trajectories of $mode_1$, (b) represents learning trajectories of $mode_2$

B. Stack Boxes

In order to further explain the application of the proposed method, the robot is shown how to stack a cubic box on another (Fig. 7), then it is required to stack four cubic boxes together in a specific order. As shown in Fig. 8, several demonstrations with different configurations of starting positions and goal positions are performed, and each demonstration has been feed to the proposed learning method. In this experiment, positions of the moving box along three axes (x, y, z) are recorded and used for trajectory learning. Motions along all the three axes are judged as discrete movements and optimal values of parameters are shown in Table III. Fig. 9 displays screenshots of one of the experiments.

IV. DISCUSSIONS

A. Pattern Analysis Discussions

This paper presents a method to automatically determine the pattern of a movement before feeding it to a DMPs-based learning framework. To discuss the situations where incorrect patterns have been used to match trajectories, we make two assumptions which are opposite to situations in the *Clean the Whiteboard* task. Firstly, in the first mode, the trajectory of the x-dimension is treated as rhythmic while that of the y-dimension is treated as discrete. Secondly, in the second mode, the trajectory of the x-dimension is treated as discrete while that of the y-dimension is treated as rhythmic. The learning and generalization results are displayed in Fig. 10 and Fig. 11 ("GDMP" denote the trajectories of generalized trajectory with incorrect pattern of DMPs). The resultant task trajectories are shown in Fig. 12 ("GT" and "DT" denote the generalized trajectory and desired trajectory, respectively). According to the results, several conclusions could be derived: (1) Even incorrect learning pattern can repeat the demonstration behavior; (2) The generalized trajectory of one dimension may look like geometrically similar to the demonstration more or less, but they are numerically quite different; (3) The resultant multi-dimensional trajectories are far away from the desired trajectories for completing the tasks; (4) Incorrect pattern judgements will result in failures of tasks in new situations.

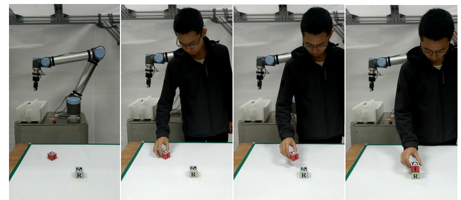


Fig. 7: Demonstration of stacking boxes

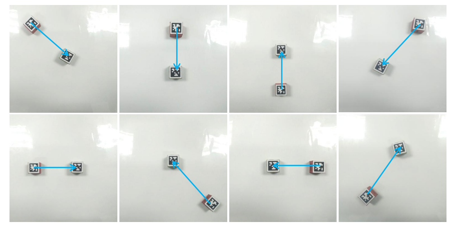


Fig. 8: Initial configurations of demonstrations

B. Cost Function Discussions

In the cost function utilized in Eq. 9, weighting factors k_i $(i = 1, 2, 3)$ are all set to one. In reality, different values can be assigned to these constants according to the control inputs of robots. For position, velocity, and torque controlled robots, relatively larger values could be assigned to k_1, k_2, and k_3, respectively.

C. Demonstration Data Discussions

To alleviate the position noise caused by the depth camera, the input trajectories used in this research are obtained by

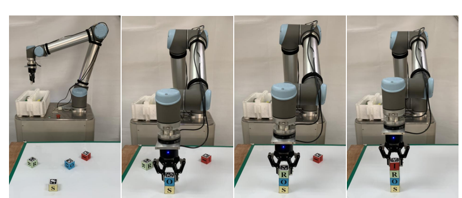
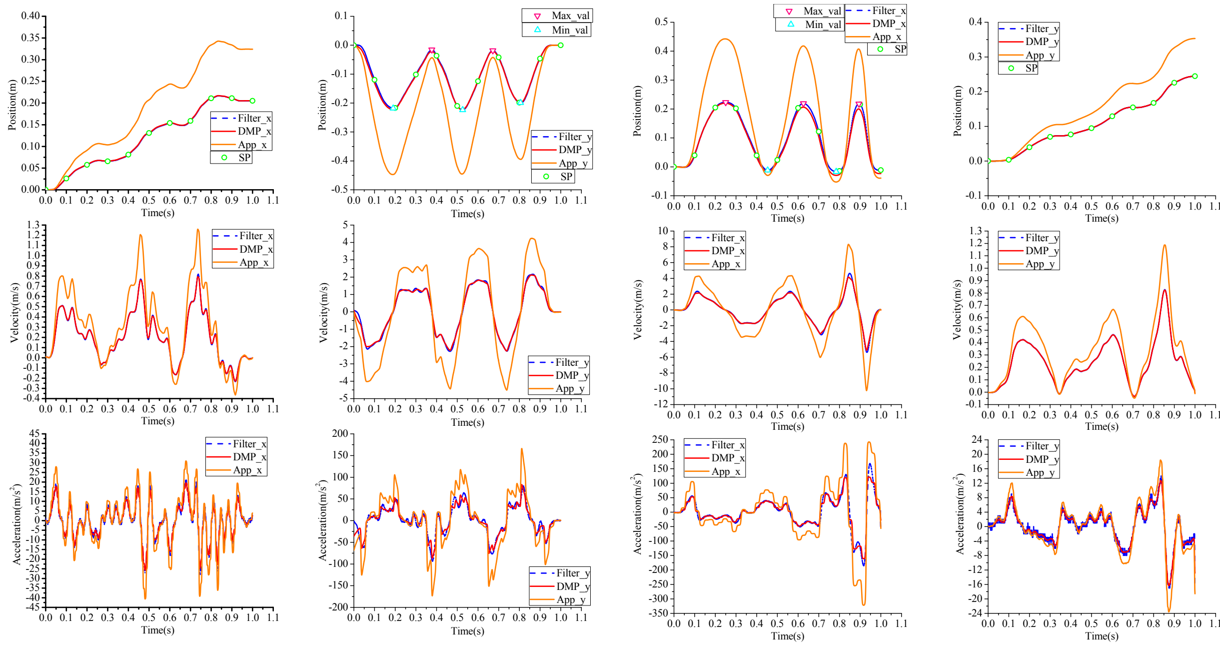
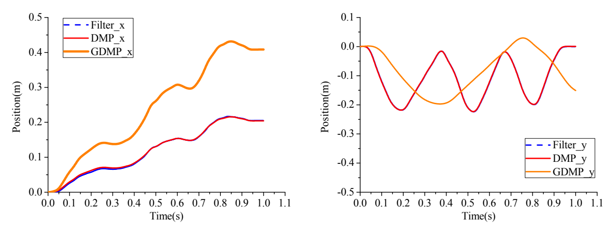


Fig. 9: Stack boxes along learned trajectories



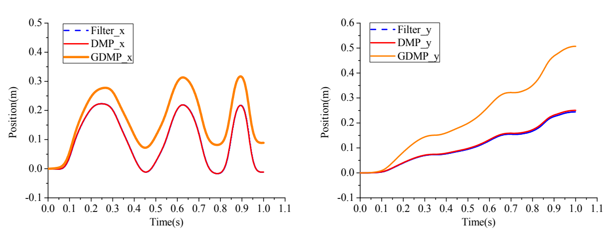
(a) (b) (c) (d)

Fig. 5: Learning and real application results of wiping demonstrations in different modes. (a) and (b) denote results of x-dimension and y-dimension demonstrated in mode$_1$, respectively, (c) and (d) represent results of x-dimension and y-dimension demonstrated in mode$_2$, respectively



(a) (b)

Fig. 10: Learning results of mode$_1$ matched by incorrect pattern. (a) x-dimension learned with rhythmic movement primitives, (b) y-dimension learned with discrete movement primitives



(a) (b)

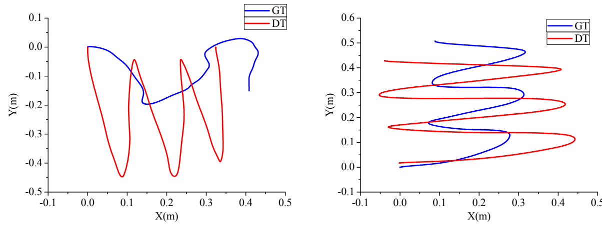
Fig. 11: Learning results of mode$_2$ matched by incorrect pattern. (a) denotes results of x-dimension learned with discrete movement primitives, (b) represents results of y-dimension learned with rhythmic movement primitives

TABLE III: Optimal parameters - Stack Boxes

Demonstration	Parameter	x	y	z
1	α_z	0.01	7.49	4.00121
	nfs	121	200	177
2	α_z	0.98	2.71	11.69
	nfs	145	124	121
3	α_z	3.69	0.01	5.24
	nfs	117	105	200
4	α_z	0.01	1.40	0.09
	nfs	143	134	159
5	α_z	0.90	11.50	41.41
	nfs	146	199	200
6	α_z	1.86	11.64	12.64
	nfs	200	199	199
7	α_z	8.17	0.01	0.01
	nfs	165	179	169
8	α_z	2.16	12.81	9.54
	nfs	200	199	200

filtering the original data. The original data can also be directly used as inputs. The value of the cost function is relatively much larger, but the proposed method can still make a correct decision of the pattern of the movement and succeed in learning the skill. In addition, A smooth trajectory can also be acquired from multiple demonstrations using Gaussian Mixture Model (GMM) [22].

Trajectories in this paper are either discrete or rhythmic. Trajectories which are composed of both rhythmic and discrete parts may not get expected results with the presented methodology. As for the DMP based recognition method utilized in this research, which is represented by Eq. 13, the recognition method based on Hidded Markov Model (HMM)



(a) (b)

Fig. 12: Resultant 2-dimensional trajectories of wiping demonstrations learned with incorrect pattern. (a) denotes resultant trajectories of mode_1, (b) represents resultant trajectories of mode_2

may also be used [23].

V. CONCLUSIONS AND FUTURE WORK

To make it easier for a non-technical robot user to programme a robot by demonstration, we present a novel methodology to obtain optimal parameters and pattern information of DMPs for unknown trajectories. The proposed method builds a direct bridge from human demonstrations to robot operations. The only thing left for a robot user to do is to provide demonstrations. Optimal parameters of both discrete and rhythmic dynamical systems can be obtained via Bayesian optimization. To ensure the reliability of learned model to be generalized to new situations in the future applications, virtual generalization behaviors are performed. The efficacy of the method is verified via experiments. Judging from the experiment results presented in this paper, the proposed method can obtain optimal parameters of both discrete and rhythmic DMPs and achieve a high success rate of pattern judgement. The method can also be used together with GMM to learn from multiple demonstrations and with movement segmentation methods [12] to learn complex tasks. Trajectories in each dimension are analyzed individually in this paper. Sometimes, more information is hidden in highly coupled multi-dimensional trajectories, which will be considered in our future research.

REFERENCES

[1] S. Schaal. "Is imitation learning the route to humanoid robots," Trends in Cognitive Sciences, 1999, vol. 3, pp. 233-242.

[2] B. D. Argall, S. Chernova, M. Veloso, et al. "A survey of robot learning from demonstration," Robotics and Autonomous Systems, 2009, vol. 57, no. 5, pp. 469-483.

[3] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, et al. "Dynamical movement primitives: learning attractor models for motor behaviors," Neural Computation, 2013, vol. 25, no. 2, pp. 328-373.

[4] A. D. Dragan, K. Muelling, J. A. Bagnell, et al. "Movement primitives via optimization," IEEE International Conference on Robotics and Automation, 2015, pp. 2339-2346.

[5] A. J. Ijspeert, J. Nakanishi and S. Schaal. "Movement imitation with nonlinear dynamical systems in humanoid robots," IEEE Interational Conference on Robotics and Automation, 2002, pp. 1398-1403.

[6] A. J. Ijspeert, J. Nakanishi and S. Schaal. "Learning rhythmic movements by demonstration using nonlinear oscillators," IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002, pp. 958-963.

[7] L. Zhijun, Z. Ting, C. Fei, et al. "Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoid-like mobile manipulator," IEEE/ASME Transactions on Mechatronics, 2017, vol. 23, no. 1, pp. 121-131.

[8] S. Qiu, W. Guo, D. Caldwell, et al. "Exoskeleton online learning and estimation of human walking intention based on dynamical movement primitives," IEEE Transactions on Cognitive and Developmental Systems, 2020, pp. 1-1.

[9] S. Niekum, S. Osentoski, G. Konidaris, et al. "Learning and Generalization of complex tasks from unstructured demonstrations," IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 5239-5246.

[10] P. Pastor, H. Hoffmann, T. Asfour, et al. "Learning and generalization of motor skills by learning from demonstration," IEEE International Conference on Robotics and Automation, 2009, pp. 763-768.

[11] G. Maeda, M. Ewerton, T. Osa, et al. "Active incremental learning of robot movement primitives," Proceedings of the 1st Annual Conference on Robot Learning, 2017, vol. 27, pp. 37-46.

[12] F. Meier, E. Theodorou, S. Schaal. "Movement segmentation and recognition for imitation learning," Proceedings of the 15th Internation Conference on Artificial Intelligence and Stotistics, 2012, pp. 761-769.

[13] J. Ernesti, L. Righetti, M. Do, et al. "Encoding of periodic and their transient motions by a single dynamic movement primitives," IEEE-RAS International Conference on Humanoid Robots, 2012, pp. 57-64.

[14] D. Pongas, A. Billard, S. Schaal. "Rapid synchronization and accurate phase-locking of rhythmic motor primitives," IEEE/RSSJ International Conference on Intelligent Robots and Systems, 2005, pp. 2911-2916.

[15] A. Ude, A. Gams, T. Asfour, et al. "Task-specific generalization of discrete and periodic dynamic movement primitives," IEEE Transactions on Robotics, 2010, vol. 26, no. 5, pp. 800-815.

[16] B. Shahriari, K. Swersky, Z. Wang, et al. "Taking the human out of the loop: a review of bayesian optimization," In Proceedings of the IEEE, 2016, vol. 104, no. 1, pp. 148-175.

[17] S. Vijayakumar, S. Schaal. "Locally weighted projection regression: an O(n) algorithm for incremental real time learning in high dimensional space," International Conference on Machine Learning, 2000, pp. 1079-1086.

[18] A. Klein. "Fast Bayesian optimization of machine learning hyperparameters on large datasets," In Proceedings of the $20th$ International Conference on Artificial Intelligence and Statistics, 2017, vol. 54, pp. 528-536.

[19] K. Yuan, I. Chatzinikolaidis, Z. Li. "Bayesian optimization for whole-body control of high-degree-of-freedom robots through reduction of demensionality," IEEE Robotics and Automation, 2019, vol. 4, no. 3, pp. 2268-2275.

[20] I. W. Selesnick, C. S. Burrus. "Generalized digital Butterworth filter desing," IEEE Transactions on Signal Processing. 1998, vol. 46, no. 6, pp. 1688-1694.

[21] J. Wang, E. Olson. "AprilTag 2: Efficient and robust fiducial detection," IEEE/IRJ International Conference on Robots and Systems (IROS), 2016, pp. 4193-4198.

[22] C. Li, C. Yang, Z. Ju, et al. "An enhanced teaching interface for a robot using DMP and GMR," International Journal of Intelligent Robotics and Applications, 2018, vol. 2, no. 1, pp. 110-121.

[23] A. B. Pehlivan, E. Oztop. "Dynamic movement primitives for human movement recognition," 41st Annual Conference of the IEEE Industrial Electronics Society, 2015, pp. 2178-2183.