Skill Learning and Inference Framework

Sang Hyoung Lee[1] and Il Hong Suh[2,⋆]

[1] Education Center for Network-Based Intelligent Robotics
[2] Department of Computer Science and Engineering, Hanyang University, Seoul, Korea
{zelog,ihsuh}@hanyang.ac.kr

Abstract. We propose a skill learning and inference framework, which includes five processing modules as follows: 1) human demonstration process, 2) autonomous segmentation process, 3) process of learning dynamic movement primitives, 4) process of learning Bayesian networks, 5) process of constructing motivation graph and inferring skills. Based on the framework, the robot learns and infers situation-adequate and goal-oriented skills to cope with uncertainties and human perturbations. To validate the framework, we present the experimental results when using a robot arm that performs a daily-life task.

Keywords: Skill learning, probabilistic affordance, effect-based clustering.

1 Introduction

Usually, in a common workspace, humans may trust a robotic system if the system shows the reliable task-dependent behaviors that the humans expect [1, 2]. To develop such a robotic system, a crucial aspect needs to be considered in which the system is able to execute necessary task skills to cope with unexpected situations dependably. Human trust can be increased if the robot is able to enhance skill functionality enough to complete its task dependably. For this, a robot should therefore be able to learn a task from human experts. It is noted here that a task usually consists of a set of motion primitives [3, 4].

To understand such motion primitives, let us consider an example, an "tea service" task. In this example, a robot performs the task using a nominal sequence as follows: The robot first delivers a cup after picking it up. Next, it inserts a teabag into the cup, after which it pours water into the cup using a kettle. Finally, a cup of tea is delivered to the human. Here, 'approaching', 'delivering', 'grasping', and 'releasing' can be considered as the skills for the "tea service" task. The motion primitives and their sequential combinations may need to be modified or changed to resolve unexpected uncertainties and perturbations while the robot performs the "tea service" task. Human trust in particular increases when the robot achieves its goals under such uncertainties and perturbations. For instance, i) if a robot can approach the objects by modifying the motion primitives of "approaching" under different initial and goal configurations, ii) if the robot can re-grasp the objects by repeating the motion primitives of "grasping" against the case that if fails to grasp the objects at its first trial, or iii) if the robot can

⋆ Corresponding author.

K.-U. Kühnberger, S. Rudolph, and P. Wang (Eds.): AGI 2013, LNAI 7999, pp. 196–205, 2013.

grasp the dropped objects by sequentially performing the motion primitives such as "approaching" and "grasping" when it drops the objects, then the human may trust the robot.

In this paper, following three research issues are investigated to resolve the crucial aspect for enhancing human trust of robot systems: learning of task skills without human interventions;

1) how to acquire task-dependent motion primitives embedded in robot tasks; the robot should be able to learn motion primitives embedded in a task by segmenting streams of motion (i.e., motion trajectories). Here, motion primitives need to be modified to guarantee their goal achievement under uncertainties and perturbations.
2) how to represent motion primitives; the robot should be able to learn the relationships between motion primitives and task-relevant entities for activating motion primitives with respect to current and target situations.
3) how to recombine motion primitives; the robot should be able to generate task-dependent behaviors by recombining motion primitives to cope with unexpected situations or unexperienced situations.

On the other hand, to validate our proposed framework, we consider a daily-life task in the following: the framework is evaluated by an experiment in which a robot provides a cup of tea for a human with "tea service" task. In this experiment, we show how well our proposed segmentation and learning of task skills works.

2 Skill Learning and Inference Framework

For this purpose, a novel autonomous segmentation method is proposed for learning motion primitives using training data obtained from imitation learning, as shown

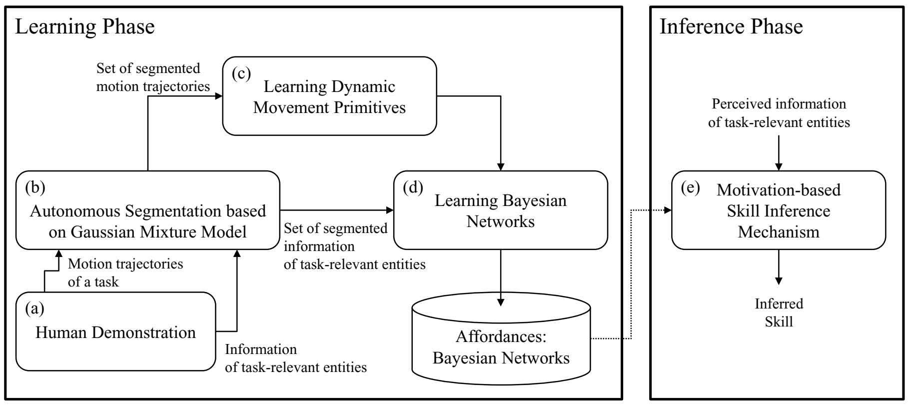

Fig. 1. Five processes for three abilities required of an skill learning and inference framework: (a) imitation process, (b) autonomous segmentation process, (c) learning process of motion primitives, (d) learning process of affordances, and (e) skill inference process

in Fig. 1(a). To date, many researchers have proposed autonomous segmentation approaches [5–7]. Even though these approaches learn unknown motion primitives, there are some constraints that are predefined or tuned, such as fixed intervals, window sizes, fixed time, and threshold values, according to the types of tasks or the motions required to learn the task-specific motion primitives. It is therefore important to autonomously learn the motion primitives embedded in such tasks without predefining and/or tuning such parameters according to the types of tasks or motions. Moreover, the motion primitives learned by the existing approaches are not modeled enough to guarantee that task-critical movements are generated, since most of existing approaches have taken note of movements that show large spatial variations [8, 9]. In many real tasks, we need to give attention to the movements that show small spatial variations, since such movements are critically essential for achieving given tasks [11]. For example, let us reconsider "tea service" task as mentioned earlier. The tasks broadly consist of three movements as follows: 1) Approaching, 2) Inserting and Pouring, and 3) Departing. Here, "Inserting" and "Pouring" movements may be not well modeled despite of their critical importance in achieving the task, because it shows small spatial variation. To learn such movements, the motion primitives should be modeled with respect to the complexity and accuracy of the motion trajectories. Our segmentation points are estimated using a Gaussian Mixture Model (GMM) with respect to the temporal and spatial complexities based on the entropies of the GMM, as shown in Fig. 1(b). We acquire motion primitives in order to generate movements that show small variations (i.e., task-critical movements) as well as large variations.

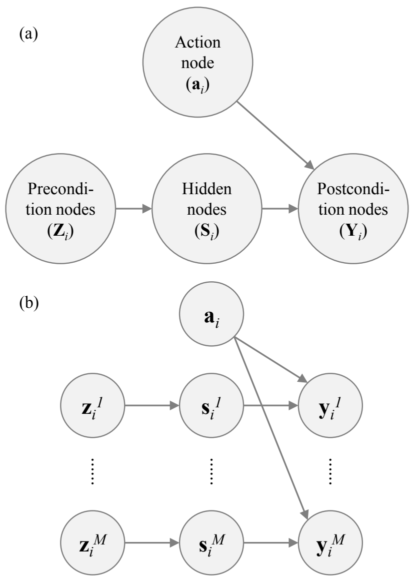

Fig. 2. Representations of an affordance: (a) causation of an affordance and (b) Bayesian network representing a probabilistic affordance

Next, the segmented motion trajectories are formalized as Dynamic Movement Primitives (DMPs) proposed in [10], as shown in Fig. 1(c). The DMPs guarantee the convergence of their goals under various uncertainties and perturbations such as changing initial and target situations as well as human intentions. Finally, to achieve the robot tasks, the motion primitives should be combined as a sequence enough to satisfy the current and target situations. To this end, probabilistic affordances are formalized to activate the motion primitives, as shown in Fig. 1(d).

The probabilistic affordances are represented as Bayesian Networks (BNs) using the information of task-relevant entities and DMPs (i.e., actions), as shown in Fig. 2. Before learning the affordances, the DMPs are clustered based on effect equivalence. The effect equivalence is a concept for the potentialities of generating a certain effect [12]. To acquire the effect equivalence, effect values are calculated by the differences between the information of task-relevant entities obtained in two segmentation points before and after executing a DMP. To cluster the training data based on effect, let us look at the training data. The training data are defined as a set of three-tuples segmented by the autonomous segmentation process. The set of three-tuples is defined as $\mathbb{T} = \{\mathbf{T}_1, ..., \mathbf{T}_N\}$. Here, N indicates the number of training data. The three-tuple is defined as

$$\mathbf{T}_i =< \mathbf{Z}_i, \mathbf{A}_i, \mathbf{Y}_i >, \tag{1}$$

where $\mathbf{Z}_i$ and $\mathbf{Y}_i$ are sets of variables that represent the configurations of task-relevant entities perceived in the segmentation points before and after the motion primitives, $\mathbf{A}_i$ acquired by the i^{th} segmentation point, and they are defined as $\mathbf{Z}_i = \{\mathbf{z}_i^1, ..., \mathbf{z}_i^M\}$ and $\mathbf{Y}_i = \{\mathbf{y}_i^1, ..., \mathbf{y}_i^M\}$. Here, M is the number of variables that represent the configurations of task-relevant entities, $\mathbf{A}_i$ is defined as $\mathbf{a}_i$, which is a variable, since a robot can execute only a single motion primitive at a time. The configurations of task-relevant entities associated with the motion primitives that have the same effects are clustered according to effect-based clustering. For this, the effect $\mathbf{E}_i$ can be calculated by various strategies as

$$\mathbf{E}_i := f(\mathbf{Y}_i, \mathbf{Z}_i), \tag{2}$$

where f is a function for calculating the effect using $\mathbf{Y}_i$ and $\mathbf{Z}_i$. In this paper, the function f is defined as operator $\ominus$, which calculates the difference by subtracting $\mathbf{Z}_i$ from $\mathbf{Y}_i$. Effect $\mathbf{E}_i$ is a set of effect values calculated as $\mathbf{E}_i = \{\mathbf{e}_i^1, ..., \mathbf{e}_i^M\} = \{\mathbf{y}_i^1 - \mathbf{z}_i^1, ..., \mathbf{y}_i^M - \mathbf{z}_i^M\}$. In addition, the effect values are substituted in the direction of the effect values, because using directions can improve the generality in comparing the similarities of effects over wider ranges. For example, let us consider an example in which a robot approaches an object from an initial position that is different from the goal position. The robot may execute the other motion primitives for approaching the object, such as stretching its arm by 20 or 30 cm. Although the motion primitives are physically different, their effects are the same (i.e., closed) when using the direction of effect values. In this case, the direction of the effect is calculated as

$$\mathbf{e}_i^j = \begin{cases} 1, & \text{if } \mathbf{y}_i^j - \mathbf{z}_i^j > 0 \\ -1, & \text{if } \mathbf{y}_i^j - \mathbf{z}_i^j < 0 \\ 0, & \text{if } \mathbf{y}_i^j - \mathbf{z}_i^j = 0 \end{cases}, \tag{3}$$

where $\mathbf{z}_i^j$ and $\mathbf{y}_i^j$ are the variables that represent the i^{th} are the variables that represent the j^{th} training data. As a result, the training data are finally clustered by exactly comparing the set of directions.

Each skill (i.e., affordance) is represented as a BN using the training data included per cluster. That is, in such BN, the parameters are learned using the training data that are clustered according to their effects. These consist of $P(\mathbf{Z}_i)$, $P(\mathbf{a}_i)$, $P(\mathbf{S}_i|\mathbf{Z}_i)$, and $P(\mathbf{Y}_i|\mathbf{a}_i,\mathbf{S}_i)$ as per the structure of Fig. 2. $P(\mathbf{Z}_i)$ and $P(\mathbf{a}_i)$ are learned as conditional probability tables or probability distributions using the frequencies of $\mathbf{Z}_i$ and $\mathbf{a}_i$ in the training data. In $P(\mathbf{Y}_i|\mathbf{a}_i,\mathbf{S}_i)$, the variables $\mathbf{S}_i$ and $\mathbf{Y}_i$can also be discrete (e.g., contact sensor) or continuous (e.g., distance sensor). In addition, $\mathbf{a}_i$ is a discrete variable that has the labels of DMPs. The BN that includes both discrete and continuous variables is referred to as a hybrid BN. The most common choice to represent a hybrid BN is the linear Gaussian distribution, in which the child has a Gaussian distribution whose mean μ varies linearly with the value of the parent, and whose standard deviation σ is fixed.

When $\mathbf{s}_i^j$ and $\mathbf{y}_i^j$ are continuous variables, the distribution of $P(\mathbf{y}_i^j|\mathbf{a}_i,\mathbf{s}_i^j)$ can be expressed as

$$\begin{aligned}\hat{m}_i^j &= P(\mathbf{y}_i^j|\mathbf{a}_i = a,\mathbf{s}_i^j) = N(a\cdot\mathbf{s}_i^j + b,\sigma^2)(\mathbf{y}_i^j)\\ &= \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{\mathbf{y}_i^j-(a\cdot\mathbf{s}_i^j+b)}{\sigma})^2},\end{aligned} \tag{4}$$

where $\mathbf{s}_i^j$ and $\mathbf{y}_i^j$ are the j^{th} hidden and post-condition variable in the i^{th} cluster, respectively, and $\mathbf{a}_i$ is a discrete variable that represents the labels of the DMPs in the i^{th} cluster (or affordance). $P(\mathbf{y}_i^j|\mathbf{a}_i = a,\mathbf{s}_i^j)$ can specify all motion primitives a, because $\mathbf{a}_i$ is a discrete variable that is handled by explicit enumeration. The parameters of (4) are individually defined as

$$a = 1, b = \frac{\sum_{i=1}^{N}(\mathbf{y}_i^j - \mathbf{s}_i^j)}{N}, \sigma = \frac{\sum_{i=1}^{N}((\mathbf{y}_i^j - \mathbf{s}_i^j) - \mu)^2}{N}, \tag{5}$$

where N is the number of training data. Finally, in $P(\mathbf{S}_i|\mathbf{Z}_i)$, $\mathbf{S}_i$ and $\mathbf{Z}_i$ can also discrete or continuous variables. When both $\mathbf{z}_i^j$ and $\mathbf{s}_i^j$ are continuous variables, the distributions are assumed to be Gaussian distributions.

The robot can infer motion primitives in a given situation using the affordances, because the affordances provide probability values for all motion primitives. However, the affordance is not suitable for accomplishing a task that requires motion primitives to be performed in sequence. To achieve a task, the robot should be able to infer a situation-adequate and goal-oriented motion primitives. For this, the affordances are arranged in a sequential structure (i.e., motivation graph). Motivation values are calculated using the affordances and a motivation value propagation algorithm based on the motivation graph.

A motivation value propagation algorithm for calculating motivation values. Here, the i^{th} affordance outputs an action probability $\hat{m}_i$ to the Motivation Value Propagation (MVP) module. The MVP module propagates the motivation values calculated by the action probabilities of affordances. The motivation value of the i^{th} MVP is defined as

$$m_i = w_i \cdot \hat{m}_i, \tag{6}$$

where w_i is the weighting value for regulating action probability $\hat{m}_i$ (here, $\hat{m}_i = \prod_{j=1}^{M} \hat{m}_i^j$) according to goal-orientedness. Note that m_i is increased or decreased from $\hat{m}_i$ by the weight w_i. Here, w_i is defined as

$$w_i = \hat{m}_i + w_{(i-1)i} \cdot d \cdot m_{(i-1)}, \tag{7}$$

where $\hat{m}_i$ is an action probability of the i^{th} affordance and $w_{(i-1)i}$ is the weighting value that represents the relationship between the $(i-1)^{th}$ and the i^{th} affordances, $m_{(i-1)}$ is the motivation value of the $(i-1)^{th}$ affordance, and d is a decay factor. This algorithm is similar to the spreading activation algorithm that propagates activation values of limited source nodes to all associated nodes based on already-learned weighting values. However, in the MVP algorithm, the motivation values are all source nodes, and the weighting values are calculated during runtime. The weight w_i is determined by the action probability and motivation value of the upper affordances. Moreover, the algorithm satisfies situation-adequateness as well as goal-orientedness [4]. This algorithm tends to increase the motivation values of reliable motion primitives and leads to comparatively more goal-oriented motion primitives in the current situation.

Thus, an intelligent robot selects dependable motion primitives based on the motivation values calculated from the probabilistic affordances as a behavior-based control approaches. The robot infers goal-oriented and situation-adequate skills based on the maximum motivation value that is calculated by the BNs and the motivation propagation algorithm, as shown in Fig. 1(e). Based on the motivation values, it is possible to generate fully connected transitions between the motion primitives without designing or learning their transition models [13]. As a result, the proposed framework increases the human trust of a robotic system by guaranteeing the achievement of the given tasks under various uncertainties and perturbations.

3 Experimental Results

To validate the framework, the "tea service" task introduced in Section 1 was tested using the Katana robot arm developed by Neuronics and twelve V100:R2 motion capture

Fig. 3. Illustrations of a robot arm and a motion capture system: (a) Katana robot arm and four objects (i.e., cup, kettle, teabag, and human hand) and (b) Experimental environment containing a robot arm and a motion capture system

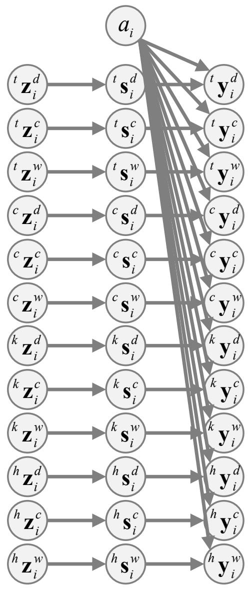

Fig. 4. Structure of the probabilistic affordances in executing the tea service task. The structure of all affordances is the same. Here, left superscripts t, c, k, and h indicate a teabag, a cup, a kettle, and a human hand,respectively, and right superscripts d, c, and w indicate the relative distance, contact, and weight, respectively. Finally, right subscripts i indicate the i^{th} affordance.

cameras developed by Optitrack, as shown in Fig. 3. The training data were extracted from 50 demonstrations using a kinesthetic teaching method. In details, five demonstrations were executed for ten different initial and goal configurations of the robot and the four entities (i.e., cup, teabag, kettle, and human hand). The information on task-relevant entities was selected as follows: 1) Weight; this was measured by the weight taken by the arm of the robot. 2) Contact; this was measured by whether the robot was in contact with an object. 3) Relative distance; this was measured by the motion-capture system to recognize the distance between the robot and the entities.

Before learning the parameters of the BNs, the training data acquired by 50 demonstrations are first segmented by the autonomous segmentation process. 700 (= 50 demonstrations $\times$ 14 segments) training data are acquired after the segmentation, since each training data is divided into fourteen segments. The 700 training data are then clustered using the effect values calculated from the information of task-relevant entities after formalizing DMPs using the motion trajectories. Even though the DMPs are clustered into 31 groups such as six [ApproachingTeabag], one [GraspingTeabag], four [DeliveringTeabag], one [ReleasingTeabag], three [ApproachingKettle], two [GraspingKettle], three [Deliveringkettle], one [PouringWater], three [PlacingKettle], one [ReleasingKettle], two [ApproachingCup], one [GraspingCup], two [DeliveringCup], and one [ReleasingCup], there are no clusters with the meanings that are in-compliant with each other. Though the training data of [ApproachingTeabag] are divided into six groups, for example, they are not included in the groups that contain different meanings. In fact,

Fig. 5. Motivation graph for executing tea service task. Here, fourteen affordances are used for achieving the task.

these meanings are attached to identify the effects, but the robot does not need to the semantics to achieve the task.

To learn affordances, the information of all entities is defined as the variables of BNs, as shown in Fig. 4. Of these variables, two (i.e., weight and contact) are defined as discrete variables, with the distance variable being continuous. The hybrid BNs were parameterized as linear Gaussian distributions. As a result, total 31 BNs were learned as the skills embedded in the "tea service" task. Quantitative results for these BNs were acquired with success rates greater than 90%. The 10% failures occur because of sensor noise, except for those escaping from the workspace of the robot arm. Finally, to generate continuous motion trajectories, the BNs were arranged using the motivation graph as shown in Fig. 5.

Several experiments were conducted to verify the skill inference mechanism under uncertainties and perturbations. That is, the robot achieve the task under the situations that the robot arm and the entities are located in different initial and goal configurations from learning phase, although the goal is the same. In particular, the experiments tried to verify the generation of various sequential combinations in the situations with various human perturbations. The additional experiments are performed as follows: 1) a human moved directly toward the cup of tea while the robot was preparing it; 2) a human delivered alternately a teabag to the cup while the robot was approaching the teabag; 3) a human snatched the teabag from the robot arm while it was approaching the cup. The robot successfully executed the task in all situations by fully-connected transitions between the BNs based on the motivation values generated by combining Bayesian inference with the motivation propagation algorithm. These additional experiments were all executed with success rates above 90% in spite of slightly different

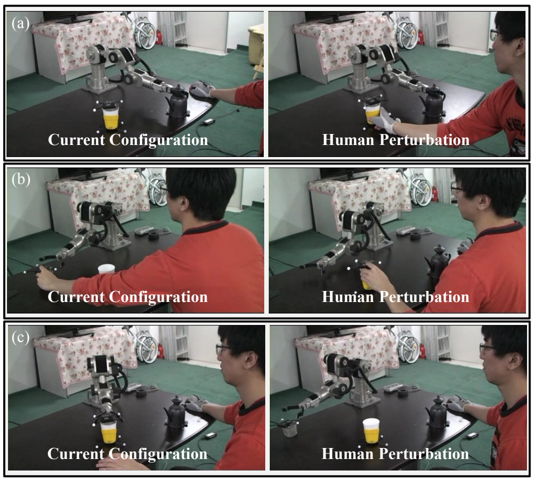

Fig. 6. Illustrations of human perturbations in the additional experiments: (a) a human directly moves to the cup while the robot is pouring water, (b) a human places the teabag into the cup while the robot is approaching the teabag to grasp it, and (c) a human snatches the teabag from the robot while it is placing the teabag into the cup.

conditions and several human interventions, as shown in Fig. 6. Moreover, the skill inference mechanism recommended skills using Bayesian inference under limited perception when some sensors (particularly the touch and weight sensors) cannot be used in the experiments.

4 Conclusion

In this paper, we proposed a unified framework for the intelligent robot to learn and infer suitable skills to achieve the given tasks reliably under uncertainties and perturbations. The main contributions of this paper are: (i) a skill learning and inference framework to cope with uncertainties and perturbations, (ii) probabilistic representation of skills using BNs, and (iii) methodology of clustering of motion primitives based on effect equivalence, and (iv) experimental evaluations using a daily-life task performed by a robot arm. In future researches, we intend to extract formal rules from the BNs to create sequences of skills for achieving novel tasks, just as humans create a number of novel sentences using words and grammar rules.

Acknowledgments. This work was supported by the Global Frontier R&D Program on <Human-centered Interaction for Coexistence> funded by the National Research Foundation of Korea grant funded by the Korean Government (MEST)(NRF-M1AXA003-2011-0028353).

References

1. Mayer, R.C., Davis, J.H., Schoorman, F.D.: An Integrative Model of Organizational Trust. Academy of Management Review, 709–734 (1995)
2. Bamberger, W.: Interpersonal Trust - Attempt of a Definition. Scientific Report, Technical University Munich (2010)
3. Hasegawa, T., Suehiro, T., Takase, K.: A Model-based Manipulation System with Skill-based Execution. IEEE Transactions on Robotics and Automation 5(8), 535–544 (1992)
4. Lee, S.H., Suh, I.H., Calinon, S., Johansson, R.: Learning Basis Skills by Autonomous Segmentation of Humanoid Motion Trajectories. In: IEEE/RAS International Conference on Humanoid Robots, pp. 112–119 (2012)
5. Kulic, D., Takano, W., Nakamura, Y.: Online Segmentation and Clustering from Continuous Observation of Whole Body Motions. IEEE Transactions on Robotics 25(5), 1158–1166 (2009)
6. Gribovskaya, E., Billard, A.: Combining Dynamical Systems Control and Programming by Demonstration for Teaching Discrete Bimanual Coordination Tasks to a Humanoid Robot. In: ACM/IEEE International Conference on Human-Robot Interaction, HRI (2008)
7. Muhlig, M., Gienger, M., Steil, J.: Human-robot Interaction for Learning and Adaptation of Object Movements. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4901–4907 (2010)
8. Kulic, D., Takano, W., Nakamura, Y.: Incremental Learning, Clustering, and Hierarchy Formation of Whole Body Motion Patterns Using Adaptive Hidden Markov Chains. The International Journal of Robotics Research 27(7), 761–784 (2008)
9. Grimes, D.B., Rashid, D.R., Rajesh, R.P.N.: Learning Nonparametric Models for Probabilistic Imitation. Advances in Neural Information of Processing Systems 19, 521 (2007)
10. Pastor, P., Kalakrishnan, M., Chitta, S., Theodorou, E., Schaal, S.: Skill Learning and Task Outcome Prediction for Manipulation. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 3828–3834 (2011)
11. Lee, S.H., Han, G.N., Suh, I.H., You, B.J.: Skill Learning using Temporal and Spatial Entropies for Accurate Skill Acquisition. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1315–1322 (2013)
12. Sahin, E., Cakmak, M., Dogar, M.R., Ugur, E., Ucoluk, G.: To Afford or Not to Afford: A New Formalization of Affordances Toward Affordance-based Robot Control. Adaptive Behavior 15(4), 447–472 (2007)
13. Lee, S.H., Suh, I.H.: Motivation-based Dependable Behavior Selection Using Probabilistic Affordance. Advanced Robotics 26(8-9), 897–921 (2012)