深層学習によるサブタスクの連続生成による複合タスクの実行
Generating Subtasks to Complete a Compound Task using Deep Learning

5117EE02-5 Kei KASE Supervisor: Prof. Tetsuya OGATA

Abstract - It is a necessity for robots to be able to perform various tasks in unspecified environments for them to extend their application to a more general environment. Therefore, robot manipulation researches use deep learning for its generalization abilities and demonstrated the improvement of performance of visuomotor learning by multi-task learning. In the sense of multi-task learning, many of the general tasks can be broken into shorter subtasks and may be considered as compound tasks. Motivated by this observation, this work proposes a predictive visuomotor multi-task learning framework with the ability to generate subtasks to complete the target task. The framework consists of an autoencoder for obtaining visual features and a recurrent neural network (RNN) with the constraint on the hidden states to create a model of a pseudo state machine for visuomotor control. Since the constraint requires a minimum configuration, the framework can be used to train different tasks with different robots without an effort. With the proposed framework, a single RNN network can successfully learn multiple subtasks, and two robots are able to execute their target tasks.
Keywords - Predictive Learning, Compound Task, Multi-task Learning, Visuomotor Control, Neural Networks

1 Introduction

The methods of robots manipulation using deep learning are gaining prevalence since controlling robots in dynamic real-life environments is challenging with traditional manipulation methods [3]. As many of the general tasks in a real-life environment can be considered as compound tasks and it is possible to define the compound tasks with shorter subtasks. That said, robots with the ability to learn multiple subtasks and generate them at an appropriate situation to complete the target task is beneficial. A state machine is one of the traditional approaches for robots to execute multiple subtasks. The robot control by a state machine can choose a specific subtask for a particular input and embed error recovery for a robust task generation. The state machine fit for a specific task may be implemented by deep learning; however, the prepared subtasks are often hand-engineered motions [1]. Although the subtasks can be also learned using deep learning for its beneficial aspects of generalization ability, the function of choosing the subtasks and executing subtasks are done by two different models.

As the scheme of visuomotor control using deep learning is gaining attention, various tasks were trained and executed. Additionally, previous works have reported that multi-task learning tasks improve the overall performances [2]. Nevertheless, multi-task learning may be applied to the learning of a compound task from shorter subtasks. Inspired by the characteristics of compound tasks, state machines, and multi-task learning, this work focuses on multi-task learning subtasks of a target task, and visuomotor control robots to execute learned subtasks consecutively to complete the target task with a deep learning framework with a state machine like characteristic.

We propose a predictive learning framework which learns multiple subtasks and generates them consecutively to complete the target tasks. The proposed framework is composed of two neural networks for visuomotor control in end-to-end fashion. The first neural network is an autoencoder (AE) which extracts visual features autonomously from images captured from a robot's camera. The second neural network is a recurrent neural network (RNN) which uses integrated information of the current visual features and the current robot state to predict the next step. For an RNN to generate task consecutively, the constraint is implemented to the hidden states to embed characteristics of a state machine to the RNN.

2 Proposed Method

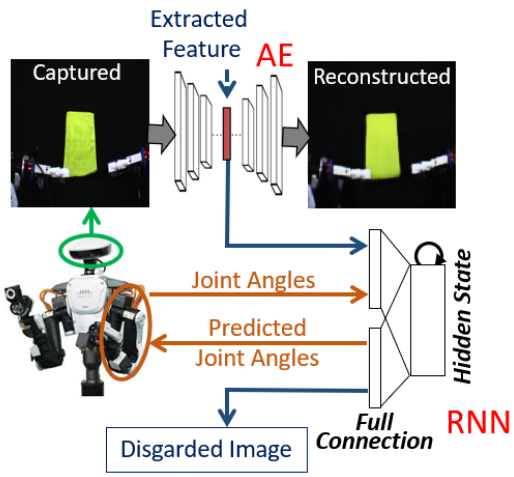

Fig. 1 DNN Framework for Visual Motor Control

In this section, we present our proposed method of the predictive learning framework with the characteris-

tics of the state machine as shown in Fig. 1. The framework uses two types of Deep Neural Networks (DNN), an autoencoder (AE) to process the visual input from the robot camera and an RNN to predict the next step of image features and robot state from previous steps. The AE is a network that outputs the same value that was used as its input. By using a bottleneck structured AE, it is possible to autonomously reduce the dimensionality of image data and use it as image features. The RNN is a network with the recurrent hidden states which act as a memory as the hidden states update their value from the current input. The RNN uses the hidden states and the current input to compute the next step. For an RNN to generate multiple subtasks consecutively to complete the target task, a constraint is applied to hidden states of RNN so that representations at the end of a certain subtask and start of the following subtask is similar. The constraint should create attracting points between each subtask and at the attracting point, the subtask is switched according to the input, emulating the characteristics of the state machine. The RNN with the constraint functions as a pseudo state machine but the state machine is "pseudo" as there is a possibility of transition to undefined states from unexpected inputs. The constraint is achieved through the calculation of the loss function shown in

$$loss = \sum_{s=0}^{S}\sum_{t=0}^{T}\|(\hat{\boldsymbol{y}}^{(s)}(t) - \boldsymbol{y}^{(s)}(t))\|^2 + \sum_{\forall(p,q)}(\gamma_{p,q}\boldsymbol{H}_p(T) - \boldsymbol{H}_q(0))^2, \tag{2.1}$$

where T is the total number of steps in a task, $\hat{\boldsymbol{y}}^{(s)}(t)$ and $\boldsymbol{y}^{(s)}(t)$ is the training signal and predicted output respectively for the sth sequence, S is the number of all sequence, $\boldsymbol{H}_p(t)$ and $\boldsymbol{H}_q(t)$ are values of hidden state for the sequence p and q, and $\gamma_{p,q}$ is the parameter that controls the loss of contexts, which is set to 1 when $(p,q)\in E$ and set to 0 when $(p,q)\notin E$ in our model. E is any set of sequence to be constrained.

3 Experiment

3.1 Task Design

Evaluation task is designed to experiment whether the framework can learn multiple subtasks and generate them in series to complete the target task. A two-armed industrial robot with 6 Degrees of Freedom at each arm named Nextage is trained to execute a compound task of towel rolling task. The towel rolling is broken into four main subtasks: (1) go to a ready pose, (2) relocate the towel, (3) roll and (4) go to a finishing pose. The subtasks are constraint so that (1) → (2), (2) → (3), (3) → (4) and (4) → (1) are connected together. If the towel does not need relocation and is already ready to roll from the beginning, the constraint is added to connect (1) → (3) as shown in Fig. 2. The AE reduces the 128×128 RGB image data into 40 dimensions. The extracted features, 12 joint angles of the left and right arms, and 2 signals for the grippers were concatenated and input to the RNN for prediction of the next steps. Furthermore, another robot named Baxter is used to learn a skewing task to show the applicability of this framework.

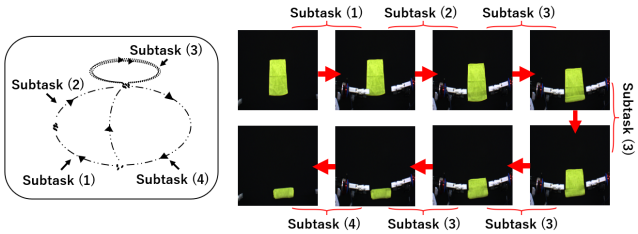

Fig. 2 (Left) Visualization of psuedo state machine, (Right) Robot view of towel rolling task.

3.2 Results

With this framework, Nextage and Baxter could connect the subtasks together to complete the target task. The visualization of the hidden state by principal component analysis for the towel rolling task is shown in Fig. 3 and the state transitions between the subtasks are observed. Furthermore, Nextage shows generalization ability against position and interruption by humans.

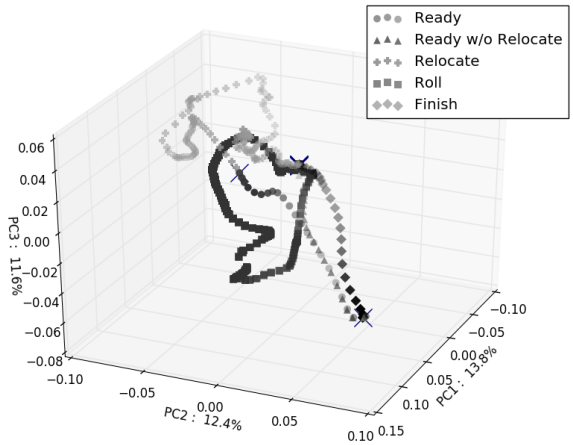

Fig. 3 Visualization of Hidden States of RNN by PCA (PC1 - PC2 - PC3)

4 Conclusion

This work proposes a method of multi-task learning of subtasks and generate them in series. The method of autonomously breaking a compound task into primitive tasks needs to be investigated in the future work.

References

[1] D. H. Grollman and O. C. Jenkins. Incremental learning of subtasks from unsegmented demonstration. *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, pages 261–266, 2010.

[2] L. Pinto and A. Gupta. Learning to push by grasping: Using multiple tasks for effective learning. *Proceedings - IEEE International Conference on Robotics and Automation*, pages 2161–2168, 2017.

[3] V. Villani, F. Pini, F. Leali, and C. Secchi. Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications. *Mechatronics*, 55(June 2017):248–266, 2018.