Teaching a Robot Pick and Place Task using Recurrent Neural Network
Giovanni de Magistris, Asim Munawar, Phongtharin Vinayavekhin

To cite this version:
Giovanni de Magistris, Asim Munawar, Phongtharin Vinayavekhin. Teaching a Robot Pick and Place Task using Recurrent Neural Network. ViEW2016, Dec 2016, Yokohama, Japan. hal-01426846

HAL Id: hal-01426846
https://hal.archives-ouvertes.fr/hal-01426846
Submitted on 5 Jan 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers. L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Teaching a Robot Pick and Place Task using Recurrent Neural Network

O Giovanni De Magistris†, Asim Munawar†, Phongtharin Vinayavekhin†
†: IBM Research - Tokyo, Tokyo, Japan
giovadem@jp.ibm.com, asim@jp.ibm.com, pvmilk@jp.ibm.com

Abstract
Programming a robot to perform a specific task is generally time consuming. This paper proposes a novel method to teach new task to a robot. The main contribution is the idea of building a task planner based on a Recurrent Neural Network (RNN). The neural network learns how to plan a task from observing a task sequence generated from a general motion planner. The method is evaluated by teaching pick and place tasks to a Baxter robot. The experiences are performed in a physical simulator. It shows that the robot can adapt to pick and place an object in various initial positions.

1. Introduction
Robots are extensively used in production lines because they can help increasing in both production speed and consistency. They also save workers from tedious and dull repetitive tasks. However, when a production line is set to a new task, significant effort is required to program each robot for this new process. More specifically, human operators have to position the robot in all the desired configurations, record these values, and then control the robot to repeat this trajectory/path during the later execution. It is also required that all objects involved are placed precisely at the same positions.

Considering the significant progress in machine learning, this human effort can be completely automated to create a robust model that can be generalized for different object types and positions. In this way, we can simplify the establishment of the production line and at the same time gain flexibility for the operation of the line.

The paper proposes a deep learning based approach to learn a pick and place task for an industrial manipulator robot. The approach can be generalized to various object initial locations. The technique is biologically inspired. Human neo cortex is essentially a prediction system that issue motor commands based on the current context and the visual and motor feedback (Fig. 1). The system uses a memory based learning system to learn the spatial and temporal encodings of an action. During the inference phase the system can predict the optimal next action to successfully complete the given task.

In this paper we have used a recurrent neural network (RNN) to teach the robot to perform the pick and place task. The system learns to take the next best action based on the previous actions and observations that it has made. Due to the feedback at each step the system can correct itself in case of minor changes during the task.

Keywords: Machine Learning, Robotics, Pick and Place

Figure 1: Human neural system decides the next action it needs to take based on the previous observations.

We give the joint angles and the object location to the system at each step and the system learns the spatial relationship between the object and robot angles. Implicitly it is learning the inverse kinematics of the robot
to perform a certain operation.

The remaining paper is organized as follows. Sec. 2 discuss the related work. The description of the pick and place task is given in Sec. 3. In Sec. 4 we illustrate the method to perform the task. Results are given in Sec. 5. Conclusions and some future directions are given at the end of the paper.

2. Related work

Over the last few decades, there have been great improvements in industrial robot programming. Early methods required to control the robot movement defining the joint trajectories. Now the programming rather performed at the higher task level [1]. At the task level, a common approach is to acquire various key-points that represent the task and generate the program accordingly. A limitation of this approach is that if anything in the recorded configuration of the environment has been modified, for instance if the object position, then the learned program is no longer valid.

To make the robot more flexible, it is becoming common to train the robot instead of explicit programming. Some research in this area use learning from demonstration to teach the robot to perform pick and place without programming to perform the task [2].

Another technique is to make the robot learn from its own experience using reinforcement learning. Deep learning combined with reinforcement learning is used to solve the pick and place task without explicit programming. Recent progress in using deep learning for robotics [3,4], an attractive idea is to develop a learning system that can be generalized to various environment conditions. In particular, this paper focuses on the change of object position of the pick and place task.

Figure 2: Images of the the pick and place states taken during Baxter performing the task: the whole task is divided and learned (using two LSTM) into two parts: Pick and Place
Although, continuous feedback visual servoning is quite common, most of the research done use networks without any memory. Deep networks with memory have several advantages. They are less prone to noise. They also remember which phase they are in right now. Therefore, instead of going directly to the target such systems can also take a sub-optimal action to follow a predefined motion trajectory to accomplish the task. In this paper we show how a memory based deep network can be used to teach the pick and place task to an industrial robot.

3. Task Description

A pick and place task is simulated in this paper to show the promising results of our approach. We perform this task using the left arm of the robot. The end-effector is the left arm gripper.

We divide the pick and place into two sub-tasks and we break down them into 5 states. The Finite State Machines of PICK and PLACE sub-tasks are shown in Fig. 2.

The states of the PICK are (the description of the Place states is similar):

- **Start**: at the start of the physics simulation, Baxter’s left arm is at random 3D end-effector position \((x_{\text{end}}, y_{\text{end}}, z_{\text{end}})\).
- **Approach**: the end-effector moves to the object position in x and y directions and \((z_{\text{obj}}+h)\) in z direction.
- **Grasp**: the end-effector goes to the object position.
- **Close Gripper**: close the gripper attached to the left arm.
- **Retract**: the end-effector moves to \((z_{\text{obj}}+h)\) in z direction.

The place position in Fig. 2 is the same for all simulations (training and test data).

4. Data-driven Task Planner using LSTM

Robot learns how to plan the task using a specific type of Recurrent Neural Network (RNN) called Long Short-Term Memory (LSTM) [5]. Two LSTM networks are trained to learn a pick task and a place task separately.

Once learned, the network is used to plan the task by varying the initial object location and robot joint positions. This section starts by explaining how training data is prepared. Then the details of how LSTM network is used to learn the task is given in the later subsection.

4.1 Preparing Training Data

Training data is generated in a physics-based virtual environment. A motion planning framework is used to create multiple samples of pick and place motion. Although each sample of pick and place sequences are continuously generated in practice, they are divided and stored separately into two training data for each task.

Each training sample for a pick task is a 10x5 matrix. Each column representing a state in a task description. The first seven dimensions represent the joint angles of the robot (left) arm, while the remaining three values are the object location. The first column among samples are randomly initialized which correspond to random end effector position and object location.

Similarly, in a place task each training sample is a 10x5 matrix and each column represents joint angles of the robot arm and object location. During data generation, the first column is copied from the last column of pick task. Although the place location \((x_{\text{place}}, y_{\text{place}}, z_{\text{place}})\) is same for all samples, the last four columns of the place task are different due to the redundancy of the robot arm.

4.2 Building a Task Planner

The task planner is constructed by learning from the training data using LSTM. As mentioned earlier, a pick task and a place task is learned separately using two networks, but with the same network architecture as shown in Fig. 3. The network is trained to output the next state, i.e. when a description of state, is given as an input, the expected output would be the description of the state, i.e., the state. The loss function of a network is a mean squared error of the output and actual desired state description.

To plan both task, the initial state of the robot arm and object location is given to the trained network of a pick task. The network will output the state description stating the joint angles that the robot arm needed to move to. An interpolator is currently used in practice, while a more
complicate motion planner can also be cooperated to move between two states of joint angles if any collision avoidance is required. The procedure is repeated until the last states of a place task is achieved.

![Network Architecture](image)

5. Results

We evaluate our approach on Baxter robot in a simulated physics-based virtual environment (Gazebo [6] and Baxter Simulator SDK [7]).

The training dataset is collected over 1400 varying objects positions of pick and place. Each sample contains 10x5 of state sequences. 80% is used as training data and 20% is used for testing. The loss function is shown in Fig. 4. Some of the generated sequences are shown in the video [https://youtu.be/suCrx4l93gA](https://youtu.be/suCrx4l93gA).

![Loss function](image)

6. Conclusion and future works

In this paper, we illustrated a method to perform a robot task using LSTM. Given a FSM, the robot learns the relationship between the joint angles and object position to perform the task. We used this approach to simulate an experimental Pick and Place robot activity.

These encouraging results show the potentials of current deep learning techniques in robotics. In a near future, we plan to apply similar techniques to different industrial tasks and real robots.

References