Learning Synergies between Pushing and Grasping with Self-supervised Deep Reinforcement Learning

Presenter: 靳子灿 左可玮 王子豪 刘瀚迪 鲁逸舟

2023/4/6

Motivation and Main Problem

Why is the problem important

With such a method, there will be even fewer people needed in transportation

Solving the problem helps robots grab things and transport them with more wisdom.

Machine learning will help collecting, filtering and analyzing the data we need and is trained to help solve the problem

AncoraSIR.com

Motivation and Main Problem

Technical challenges arising from the problem

Both networks are trained jointly in a Q-learning framework, and are entirely self-supervised by trial and error, where rewards are provided from successful grasps.

In this way, our policy learns pushing motions that enable future grasps, while learning grasps that can leverage past pushes.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Motivation and Main Problem

High-level idea of why prior approaches didn't already solve

It remains unclear how to plan sequences of actions that combine grasps and pushes, each learned in isolation.

While hard-coded heuristics for supervising push-grasping policies have been successfully developed by exploiting domain-specific knowledge, they limit the types of synergistic behaviors between pushing and grasping that can be performed.

AncoraSIR.com

Motivation and Main Problem

Key insights of the proposed work

We learn joint pushing and grasping policies through self-supervised trial and error. Pushing actions are use-ful only if, in time, enable grasping. This is in contrast to prior approaches that define heuristics or hard-coded objectives for pushing motions.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Motivation and Main Problem

Key insights of the proposed work

We train our policies end-to-end with a deep network that takes in visual observations and outputs expected that takes in visual observations and outputs expected return (i.e. in the form of Q values) for potential pushing and grasping actions. The joint policy then chooses the action with the highest Q value – i.e. , the one that maximizes the expected success of current/future grasps. This is in contrast to explicitly perceiving individual objects and planning actions on them based on hand-designed features.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Problem Setting

The best control method and off-policy Q-learning

Markov decision process:

State: $s_t \rightarrow$ new state: s_{t+1}

Action: a_t

Policy: $\pi(s_t)$

Reward: $R_{at}(s_t, s_{t+1})$

Return: $R_t = \sum_{i=t}^{\infty} \gamma R_{\alpha_i}(s_i, s_{i+1})$

A greedy deterministic policy

chooses actions by maximizing the Q-function --- $Q_\pi(s_t, a_t)$,

AncoraSIR.com

Problem Setting

Learning objective

A fixed target value y_t :

$$y_t = R_{a_t}(s_t, s_{t+1}) + \gamma\, Q(s_{t+1}, \mathrm{argmax}(Q(s_{t+1}, a')))$$

To minimize the temporal difference value δ_t between Q and y_t

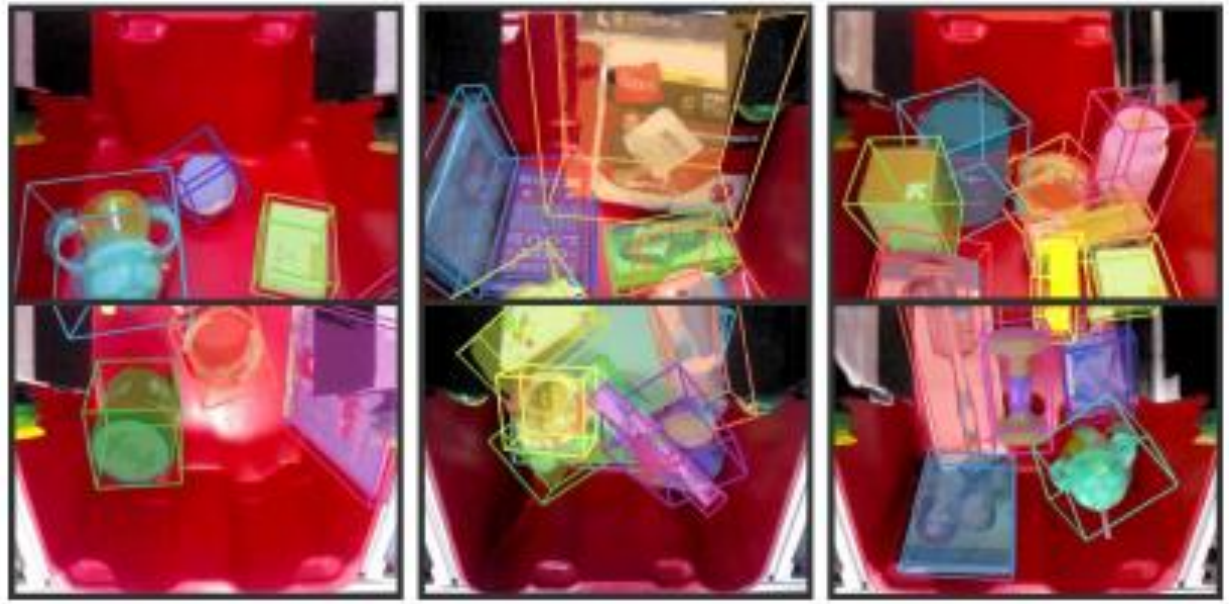
$$\delta_t = |Q(s_t, a_t) - y_t|$$

AncoraSIR.com

Limitations of Prior Work

Paper: Multi-view Self-supervised Deep Learning for 6D Pose Estimation in the Amazon Picking Challenge

1.Chaotic. Shelves and handbags may have multiple objects and may be arranged to deceive visual algorithms.



AncoraSIR.com

Limitations of Prior Work

2.Self shielding. Due to the limited location of the camera, the system can only see a partial view of the object.

3.Sensor noise. Commercial depth sensors are not reliable in capturing reflective, transparent, or mesh surfaces.

AncoraSIR.com

Proposed Method

A. State Representations

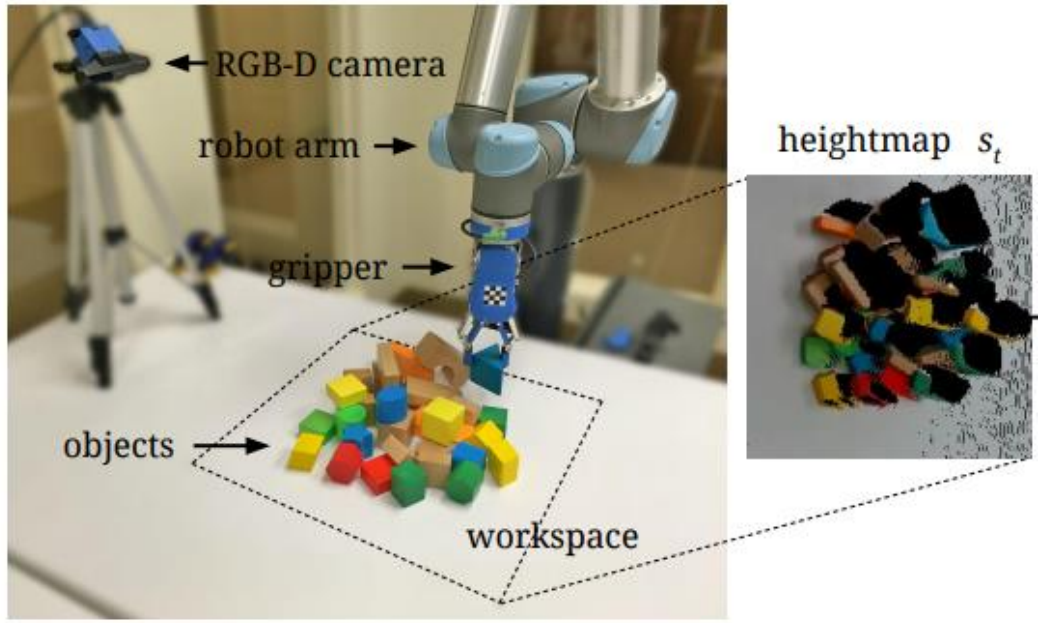
For each state s_t : visual 3D data --- heightmaps

1.Capture RGB-D images from a fixed-mount camera

2.Project the data onto a 3D point cloud

3.Orthographically back-project upwards

Workspace: $0.448m^2$
Pixel resolution: 224*224



AncoraSIR.com

Proposed Method

B. Primitive Actions

ψ : a motion primitive behavior parametrized from action α_t (can be pushing or grasping)

$$a = (\psi, q) \mid \psi \in \{\text{push}, \text{grasp}\}, q \twoheadrightarrow p \in s_t$$

Pushing: 10cm horizontal movement

Grasping: 3cm vertical decline

AncoraSIR.com

Proposed Method

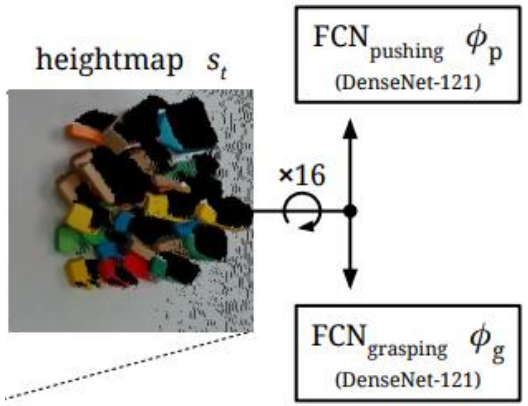
C. Learning Fully Convolutional Action-Value Functions

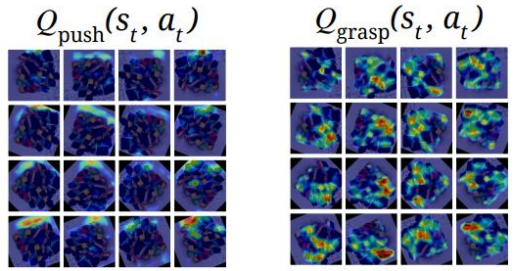
16 directions: each 22.5°

First, the Q value prediction for each action now has an explicit notion of spatial locality with respect to other actions.

Second, FCNs are efficient for pixel-wise computations.

Finally, FCN models can converge with less training data.





AncoraSIR.com

Proposed Method

D. rewards

Grasp successfully: $R_g(s_t, s_{t+1}) = 1$

(if antipodal distances between gripper fingers after a grasp attempt exceed threshold)

Pushes that make detectable changes: $R_p(s_t, s_{t+1}) = 0.5$

(if sum of differences between heightmaps exceeds threshold)

R_p does not ensure one push action enables future grasp actions!

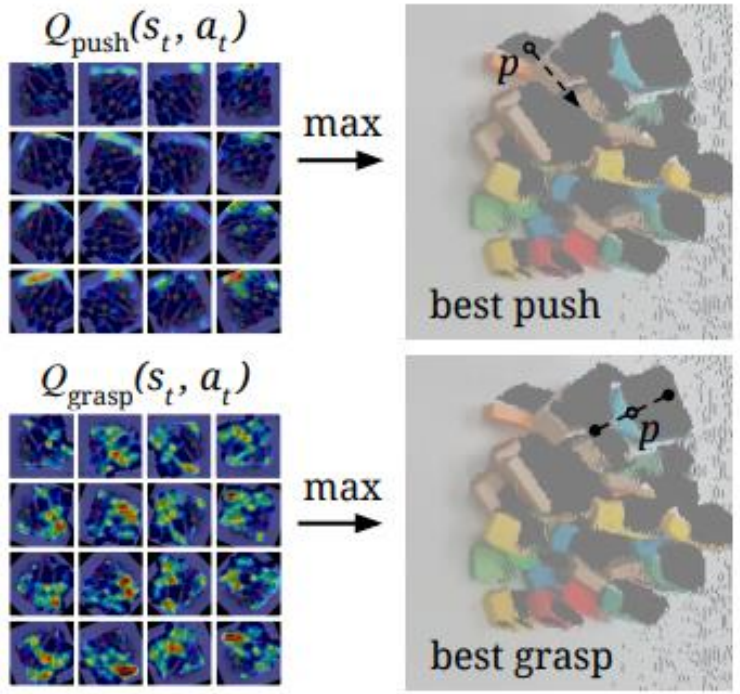
AncoraSIR.com

Proposed Method

E. Training details

Huber loss function

$$\mathcal{L}_i = \begin{cases} \frac{1}{2}(Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i^-})^2, \text{for } |Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i^-}| < 1, \\ |Q^{\theta_i}(s_i, a_i) - y_i^{\theta_i^-}| - \frac{1}{2}, \text{ otherwise.} \end{cases}$$



θ_i : parameters of the neural network at iteration I

θ_i^- : fixed

φ_ψ : network to be trained

ϵ : =0.5~0.1

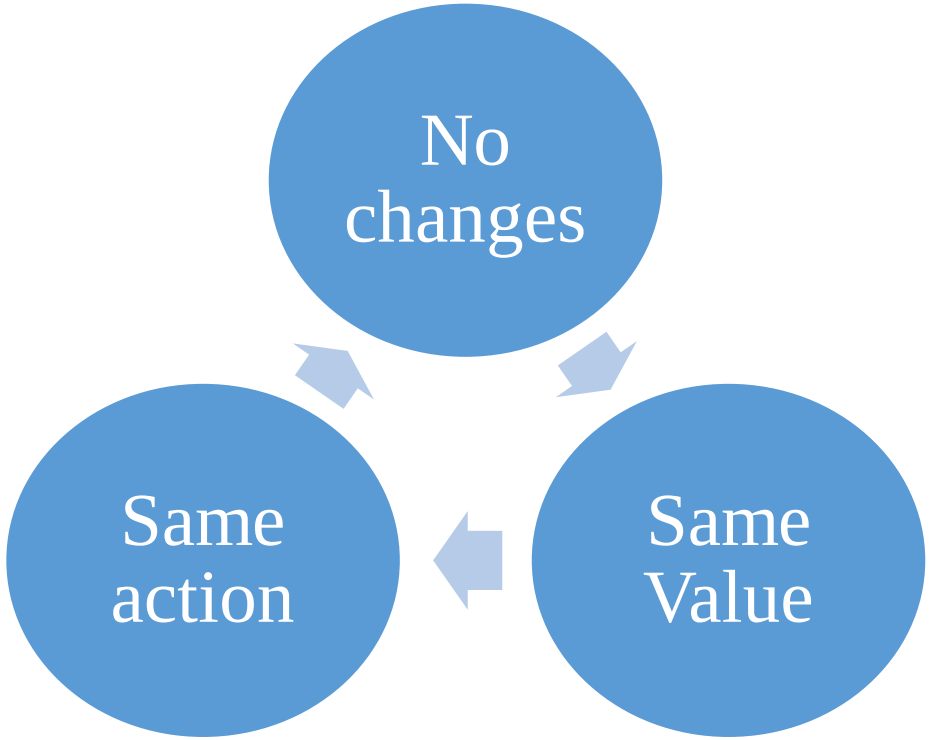
γ : future discount = 0.5

AncoraSIR.com

Proposed Method

F. Testing details

Greedy deterministic policy --- easy to get stuck (repeat same actions)



Network weights are reset to original state when:

1.All objects are grasped successfully

2.No change of environment exceed 10 times

AncoraSIR.com

Experimental Set up

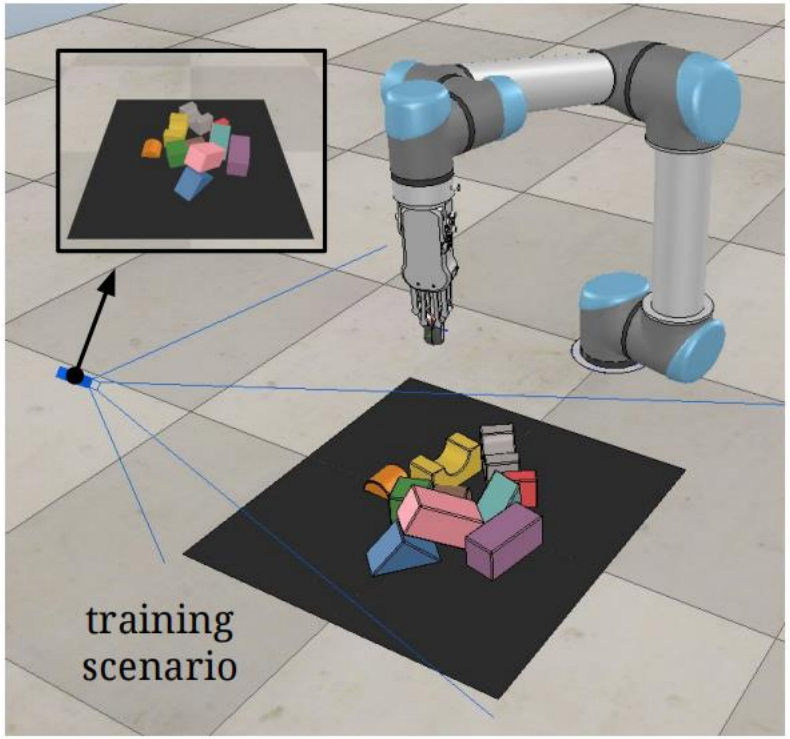
Virtual Simulation

Environment:

UR5 robot arm with an RG2 gripper in V-REP [39] (illustrated in Fig.3) with Bullet Physics 2.83 for dynamics and V-REP's internal inverse kinematics module for robot motion planning.

Each test run in simulation was run $n = 30$ times. The objects used in these simulations include 9 different 3D toy blocks

simulate a statically mounted perspective 3D camera in the environment, from which perception data is captured. RGBD images of resolution 640×480 are rendered with OpenGL from the camera, without any noise models for depth or color.



AncoraSIR.com

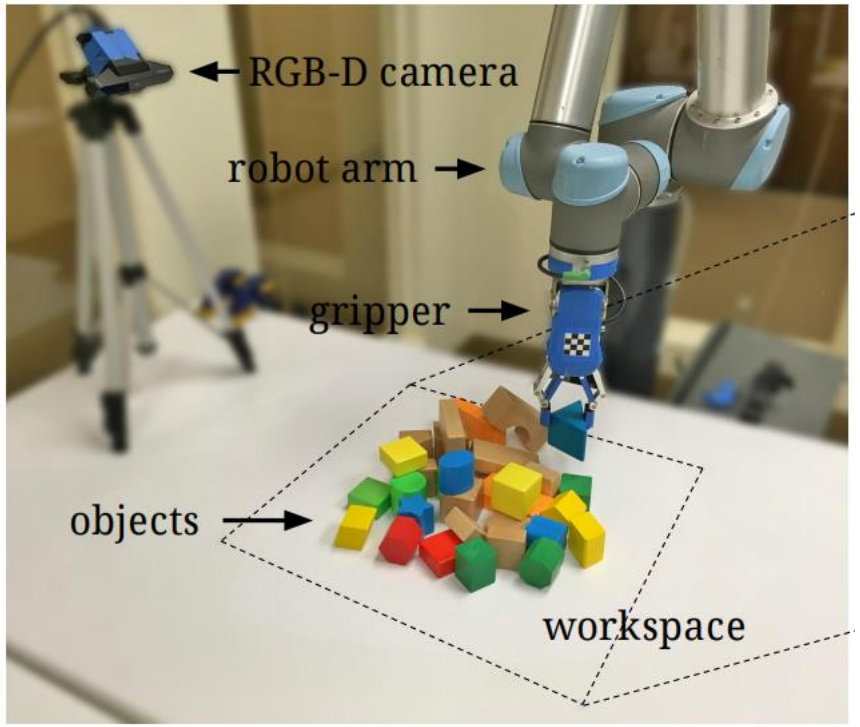
Experimental Set up

Real-world

Environment:

a UR5 robot arm with an RG2 gripper, overlooking a tabletop scenario. Objects vary across different experiments, including a collection of 30+ different toy blocks for training and testing

For perception data, RGB-D images of resolution 640 × 480 are captured from an Intel RealSense SR300, statically mounted on a fixed tripod overlooking the tabletop setting.



AncoraSIR.com

Experimental Setup

Baselines and hypotheses

Baselines:

Reactive Grasping-only Policy (Grasping-only) ;

Reactive Pushing and Grasping Policy (P+G Reactive)

Tested hypotheses:

Visual Pushing for Grasping (VPG)

AncoraSIR.com

Experimental Setup

Evaluation Metrics

Clear description of the metrics that will be used:

1) the average % completion rate over the *n* test runs, which measures the ability of the policy to finish the task by picking up all objects without failing consecutively for more than 10 attempts,

2) the average % grasp success rate per completion;

3) the % action efficiency (defined as $\frac{\textbf{\# objects in test}}{\textbf{\# actions before completion}}$), which describes how succinctly the policy is capable of finishing the task.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Experimental Results

Virtual Simulation

Random arrangements

TABLE I

SIMULATION RESULTS ON RANDOM ARRANGEMENTS (MEAN %)

Method	Completion	Grasp Success	Action Efficiency
Grasping-only [8]	90.9	55.8	55.8
P+G Reactive	54.5	59.4	47.7
VPG	**100.0**	**67.7**	**60.9**

30 objects are randomly dropped onto a table

Challenging arrangements

TABLE II

SIMULATION RESULTS ON CHALLENGING ARRANGEMENTS (MEAN %)

Method	Completion	Grasp Success	Action Efficiency
Grasping-only [8]	40.6	51.7	51.7
P+G Reactive	48.2	59.0	46.4
VPG	**82.7**	**77.2**	**60.1**

11 challenging test Cases similar to the right figure

AncoraSIR.com

Experimental Results

Success rate with respect to training steps(in simulation)

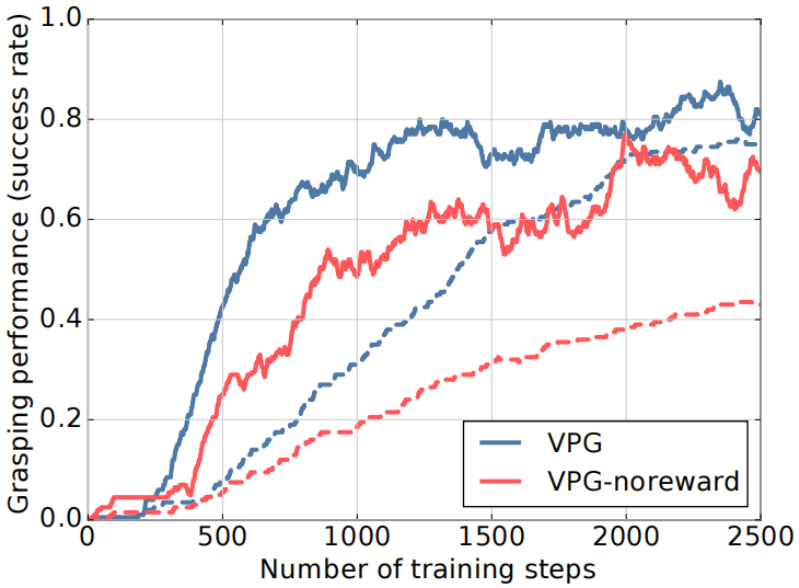


Fig. 4. Comparing performance of VPG policies trained with and without rewards for pushing. Solid lines indicate % grasp success rates (primary metric of performance) and dotted lines indicate % push-then-grasp success rates (secondary metric to measure quality of pushes) over training steps.

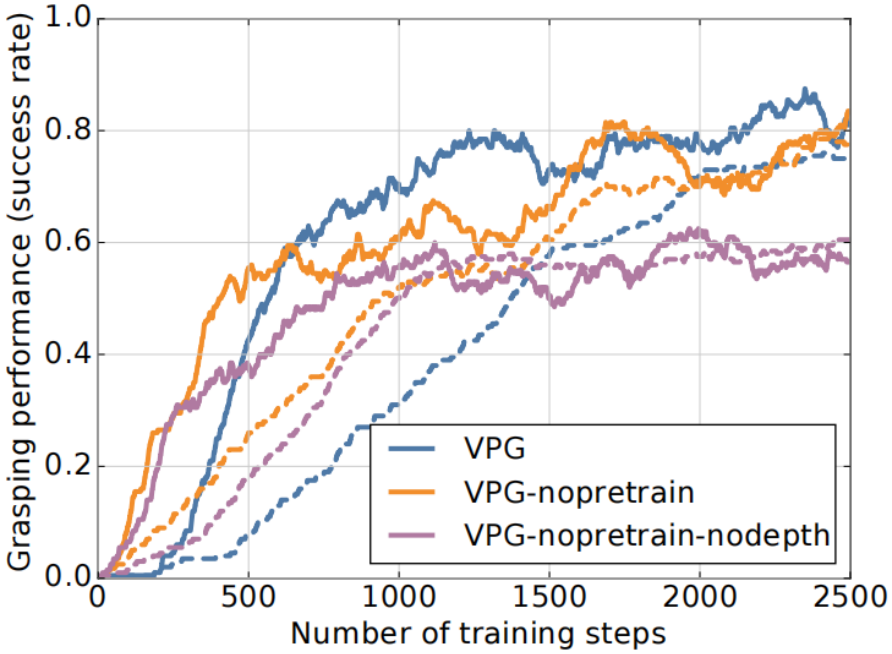


Fig. 5. Comparing performance of VPG policies initialized without weights pre-trained on ImageNet and without the depth channels of the RGB-D heightmap (*i.e.* no height-from-bottom, only color information). Solid lines indicate % grasp success rates (primary metric of performance) and dotted lines indicate % push-then-grasp success rates (secondary metric to measure quality of pushes) over training steps.

AncoraSIR.com

Experimental Results

Real-world result

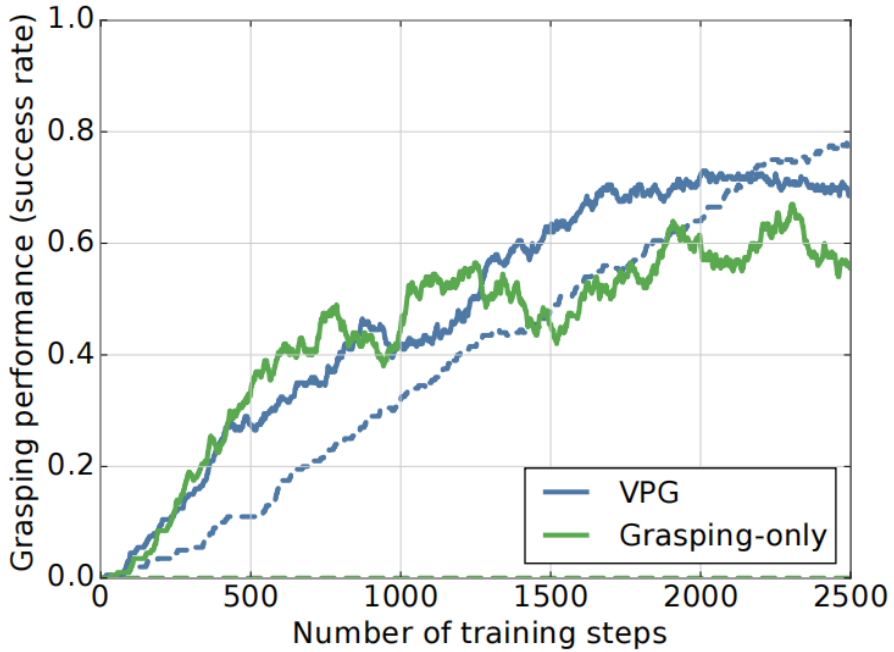


Fig. 6. Evaluating VPG in real-world tests with random 30+ object arrangements. Solid lines indicate % grasp success rates (primary metric of performance) and dotted lines indicate % push-then-grasp success rates (secondary metric to measure quality of pushes) over training steps.

TABLE IV

REAL-WORLD RESULTS ON CHALLENGING ARRANGEMENTS (MEAN %)

Method	Completion	Grasp Success	Action Efficiency
Grasping-only [8]	42.9	43.5	43.5
VPG	**71.4**	**83.3**	**69.0**

AncoraSIR.com

Discussion of Results

It can be easily concluded that VPG performs better in completion rate, success rate and efficiency.

Moreover, it has high training efficiency for it can perform well in just six hours of training.

AncoraSIR.com

Limitations

Limitations of this paper

First Motion primitives are defined with parameters specified on a regular grid (heightmap), which provides learning efficiency with deep networks, but limits expressiveness.

Second The deep learning system has only been trained with blocks and tested with a limited range of shapes (fruit, bottles, etc.).

Third The synergy just covers two examples (pushing and grasping) of the larger family of primitive manipulation actions.

AncoraSIR.com

Future Work

Ideas for future work based on limitations

First Parameterizations with more expressive motions
(parallel combinations of pushing and grasping)
Various contact surfaces

Second Training on larger varieties of shapes
further evaluate the generalization capabilities of the learned policies

Third Study strategies of many other primitive manipulation actions
(rolling, toppling, squeezing, levering, stacking, etc.)

AncoraSIR.com

Extended Readings

[1] Official website of amazon picking challenge. [Online]. Available: http://amazonpickingchallenge.org

[2] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in CVPR, 2015, pp. 3431–3440.

[3] Website for code and data. [Online]. Available: http://apc.cs.princeton.edu/

[4] R. Jonschkowski, C. Eppner, S. Hofer, R. Martin, and O. Brock, “Probabilistic multi-class segmentation for the amazon pick_x0002_ing challenge,” http://dx.doi.org/10.14279/depositonce-5051, 2016.

[5] C. Eppner, S. Hofer, R. Jonschkowski, R. Martın-Martın, A. Sieverling, V. Wall, and O. Brock, “Lessons from the amazon picking challenge: Four aspects of building robotic systems,” in RSS, 2016.

AncoraSIR.com

SUSTech
Southern University of Science and Technology

Conclusion and Summary

Summary

This paper shows that the synergy between planning non-prehensile (pushing) and prehensile (grasping) actions can be learned by the deep reinforcement learning system.

It is the first to realize performing complex sequences of pushing and grasping on a real robot in a short training time.

The planning of combined manipulation actions for robots is still a new field which has unlimited potential and needs further more research.

AncoraSIR.com

Thank You

AncoraSIR.com

SUSTech
Southern University
of Science and Technology