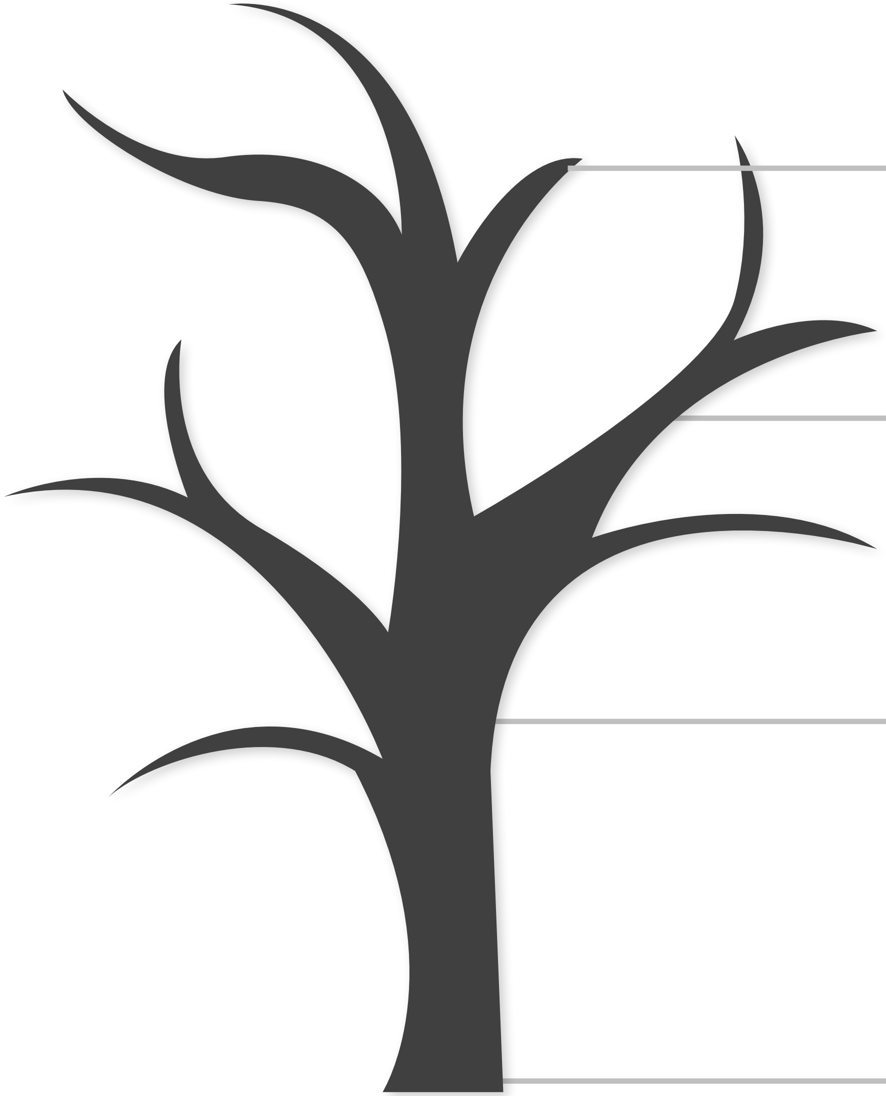
A Robotic Auto-Focus System based on Deep Reinforcement Learning

Xiaofan Yu, Runze Yu, Jingsong Yang, Xiaohui Duan*

School of EECS, Peking University
Beijing, China

Speaker: Xiaofan Yu

Outline



Background

- Passive Auto-Focus
- How to deal with auto-focus using vision input

Method

- System model
- Reward Function Design
- Deep Q Network Design

Experiments

- Hardware Setup
- Training in Virtual Environment
- Training in Real Environment

Conclusion

I. Background

Background

Passive Auto-Focus

- First and foremost step in cell detection
- Two phases in passive auto-focus techniques:
  - focus measure functions
  - search algorithms

End-to-end learning approach

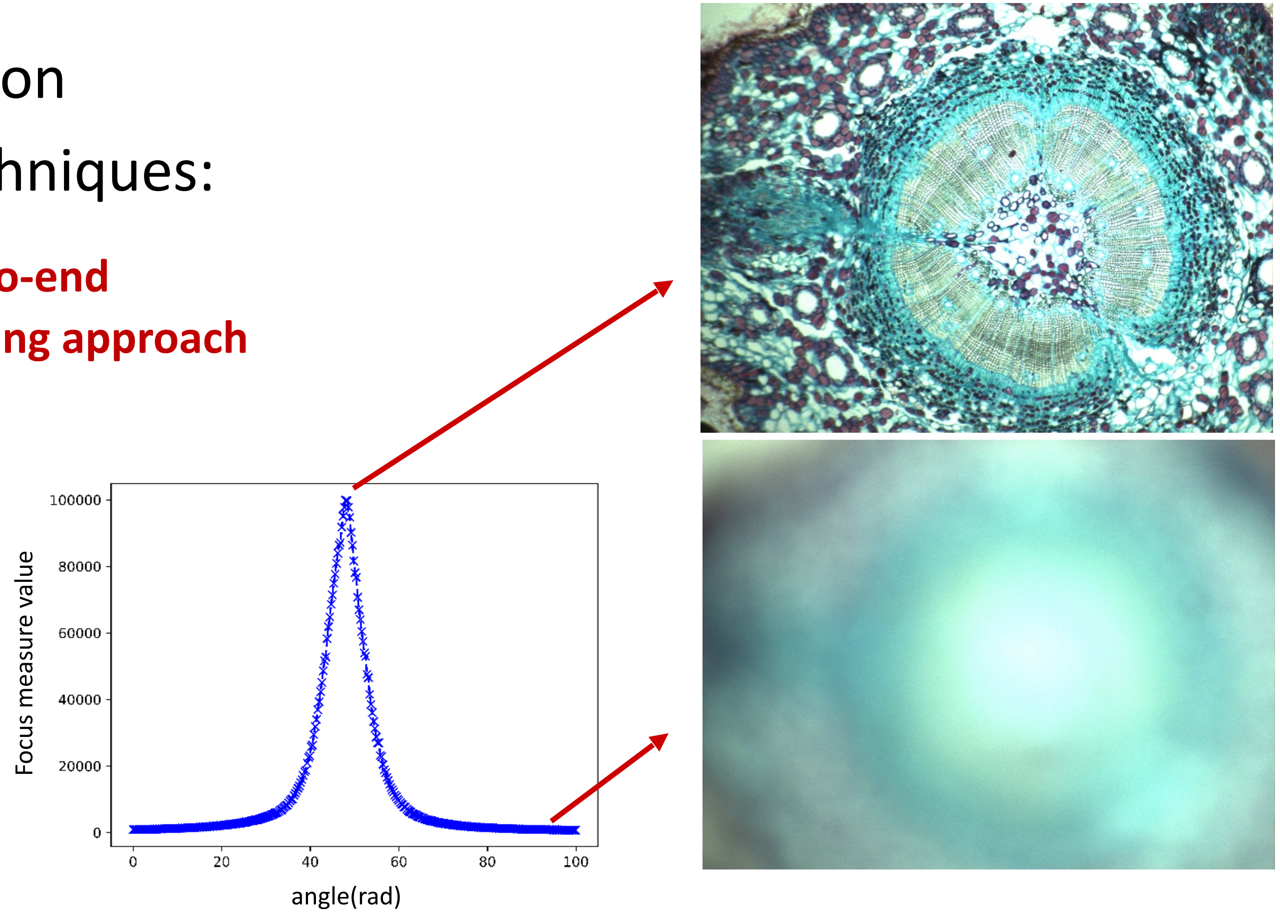


Figure 1: Mechanisms of passive auto-focus techniques.

Background

How to deal with auto-focus using vision input?

- Vision-based model-free decision-making task
- Deep Reinforcement Learning (DRL) is the solution!
  - Deep Q Network (DQN) can deal with high dimensional input

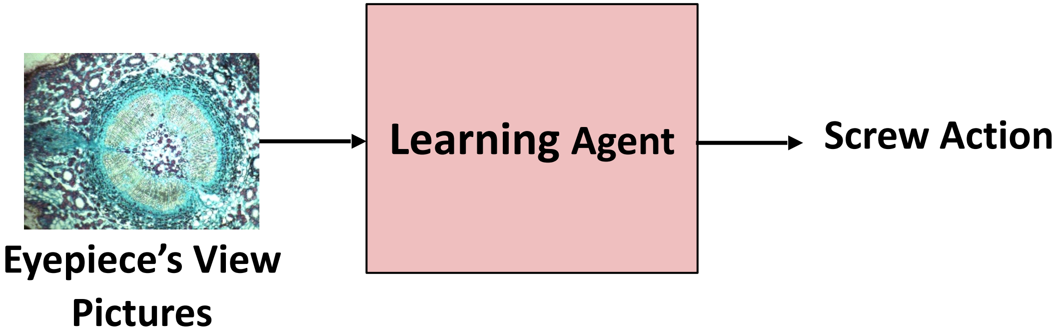


Figure 2: Model of end-to-end vision-based auto-focus problem.

Figure 3: Atari 2600 games, which could be played by DRL-trained agent with vision input [1].

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with Deep Reinforcement Learning," arXiv preprint arXiv:1312.5602, 2013.

Background

Our Contribution

- Apply DRL to auto-focus problems, which does not utilize human knowledge
- Demonstrate a general approach to vision-based control problems
  - Discrete state and action spaces
  - Reward function with an active terminal mechanism

II. Method

Method

System model

- State (s_t): three successive images (x_t) and their corresponding actions (a_t)
  - $s_t = \{x_t, a_t, x_{t-1}, a_{t-1}, x_{t-2}, a_{t-2}\}$
- Action (a_t): one in the action set
  - Action set = {coarse positive, fine positive, terminal, fine negative, coarse negative}
- Reward (r_t)
- DQN

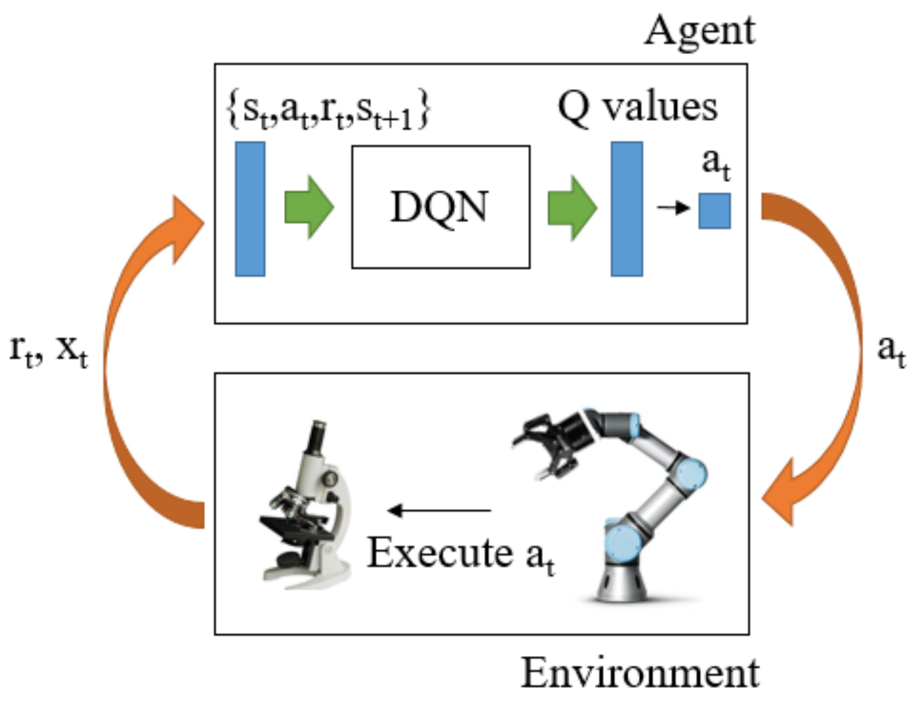
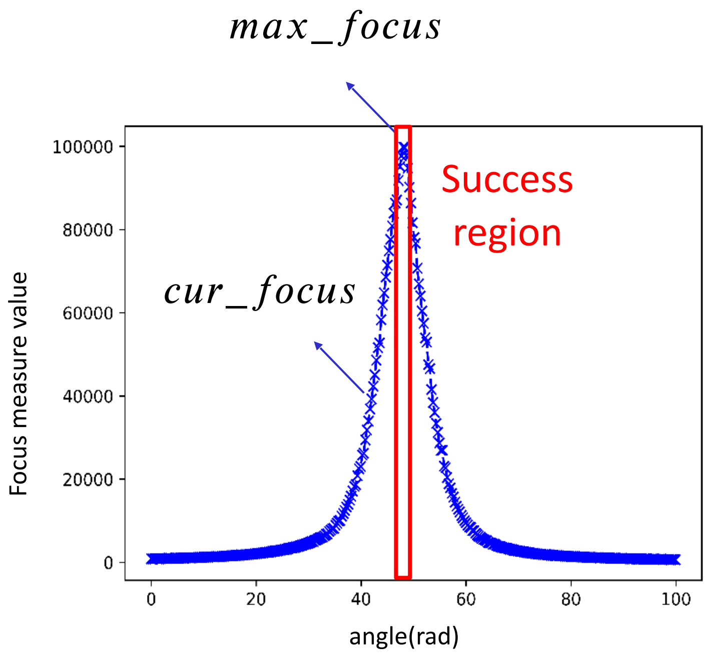


Figure 4: System model.

Method

Reward Function Design

- Reward Function
  - $reward = c \bullet (cur\_focus - max\_focus) + t$
  - c : coefficient
  - cur_focus and max_focus: current and max focus value
  - t : termination bonus, $t = \begin{cases} 100, & success \\ -100, & failure \end{cases}$



Method

■ DQN Design

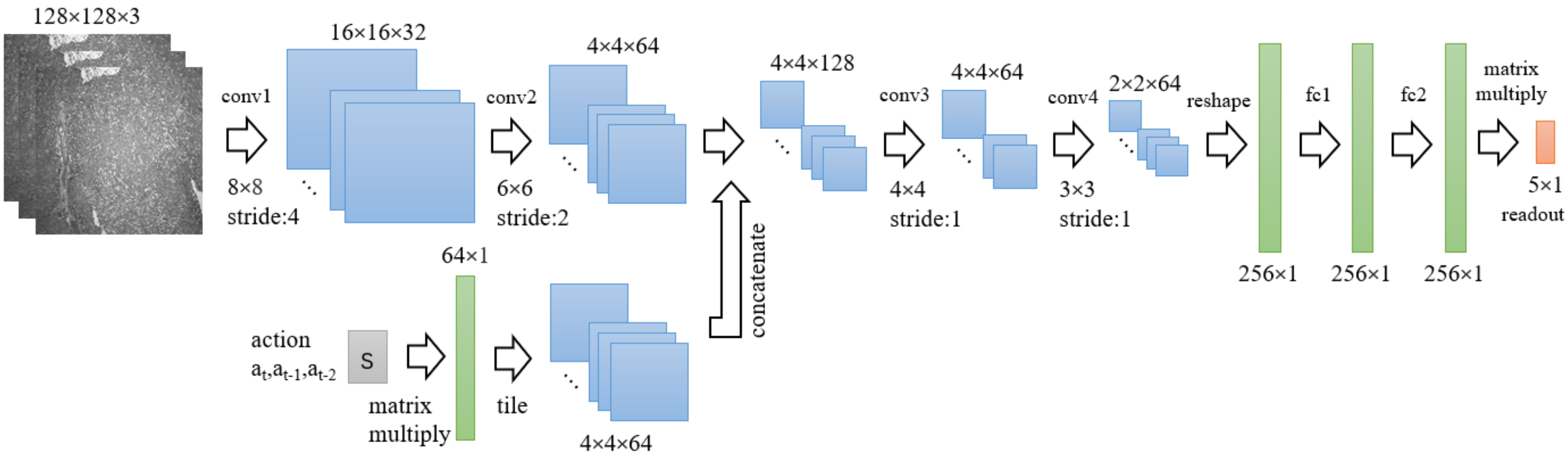


Figure 5: The architecture of our DQN.

III. Experiment

Experiment

- **Hardware Setup**
- **Training in Virtual Environment**
- **Training in Real Environment**

Figure 6: Auto-focus system implementation

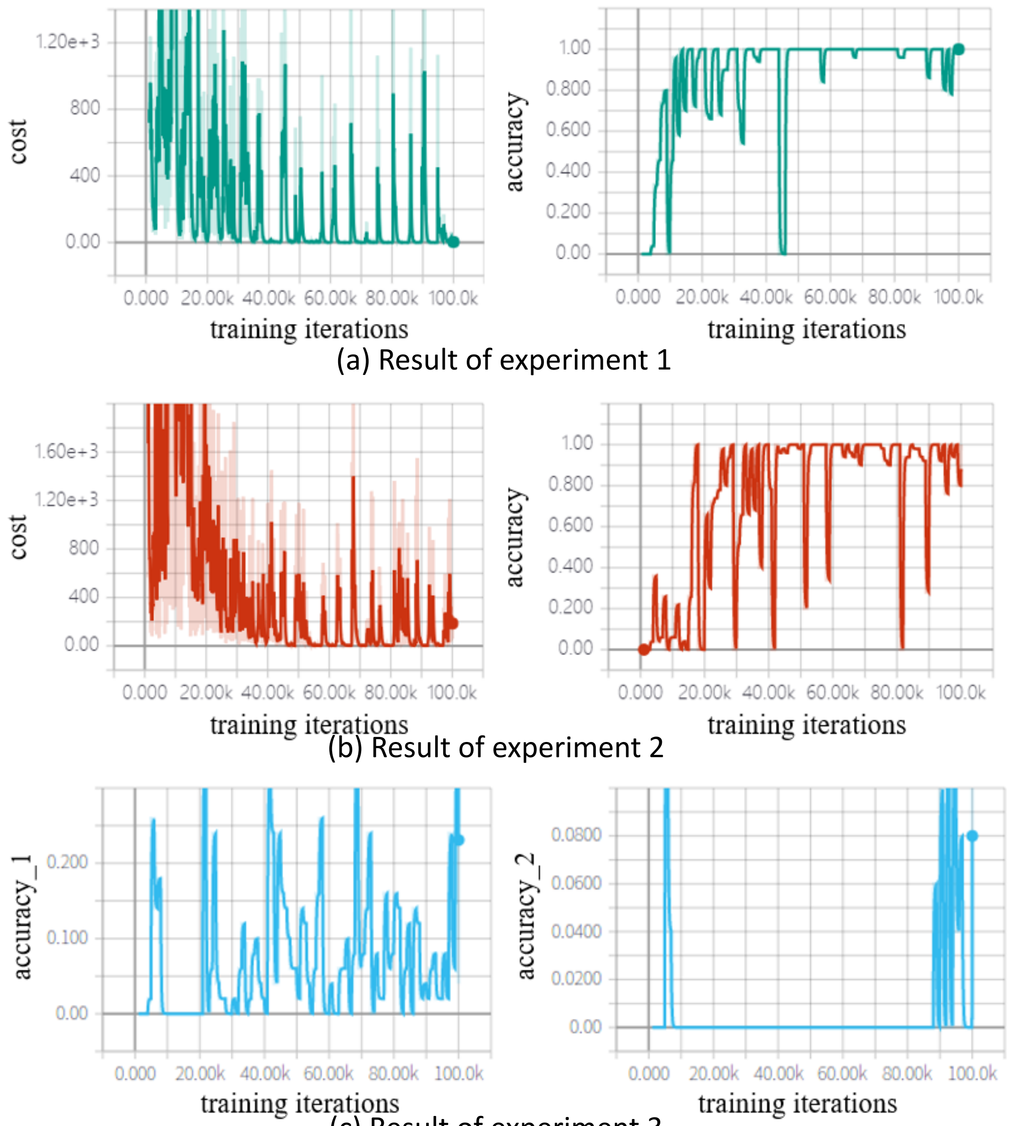
Experiment

■ Training in Virtual Environment

- Save time in real training phase
- Before training, perform equal-spacing sampling to construct a simulator

TABLE I: Experimental setups

No.	Goal	Focus Range (rad)	Train & Test Data Set
1	Basic experiment to assess the feasibility	30.0-69.0	Same view
2	Comparison experiment to assess the adaptablity to broader focus range	10.2-89.7	Same view
3	Comparision experiment to assess the adaptablity to different views	30.0-69.0	Three different views, one for training and the rest two for testing



(a) Result of experiment 1
(b) Result of experiment 2
(c) Result of experiment 3
Figure 7: Result of virtual training phase.

Experiment

■ Training in Real Environment

- Deploy the virtual-trained model to real scenarios
- Apply real training phase and obtain a new model
- Compare those two models by performing tests in real world

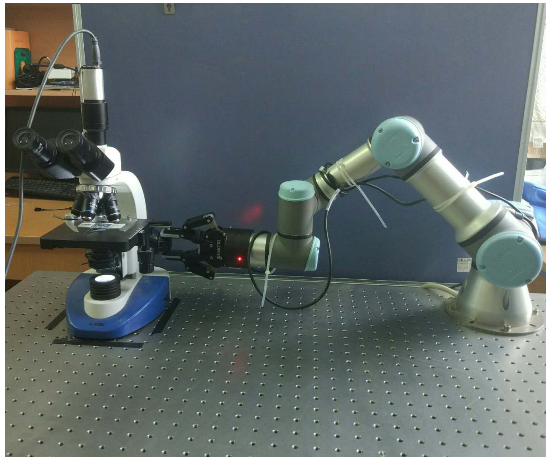


Figure 8: Real world testing scene.

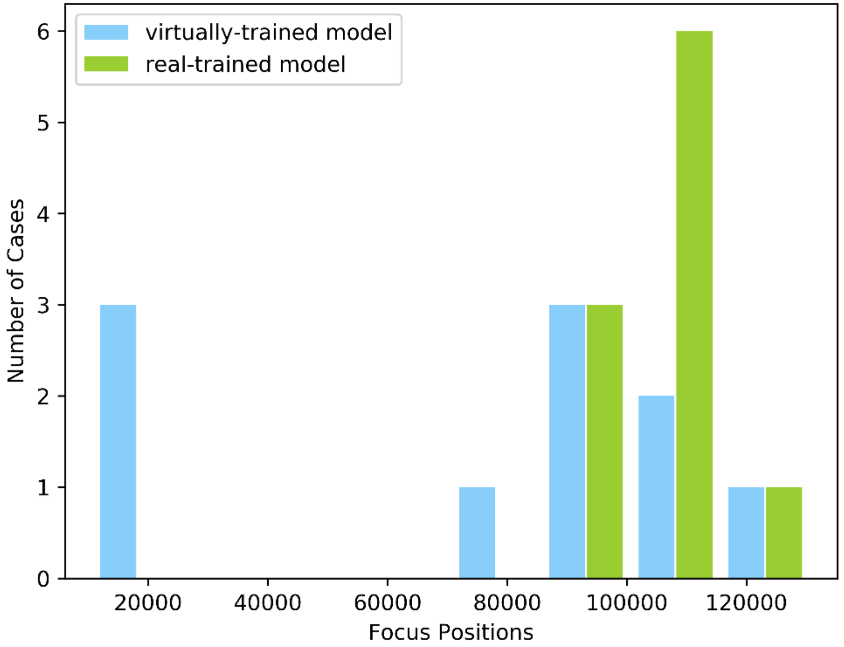


Figure 9: The histogram of focus positions.

Experiment

Summary

- In virtual training phase, our model shows great viability on larger range but need improvements on generalization capacity
- In real training phase, our method is feasible to learn accurate policies (100% success rate) in real world but is susceptible to environmental factors

IV. Conclusion

Conclusion

- **In this paper, we**
  - use DQN to achieve end-to-end auto-focus
  - demonstrate that discretization in state and action spaces and active termination mechanism could be a general approach in vision-based control problems
- **Next Step**
  - Improve generalization capacity by training with larger dataset
  - Improve robustness towards environmental factors
  - Reduce training time
  - ……

THANK YOU

Q & A

Reference

- [1] G. Saini, R. O. Panicker, B. Soman, and J. Rajan, “A Comparative Study of Different Auto-Focus Methods for Mycobacterium Tuberculosis Detection from Brightfield Microscopic Images,” in Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER), IEEE, pp. 95–100, IEEE, 2016.
- [2] J. M. Mateos-P´erez, R. Redondo, R. Nava, J. C. Valdiviezo, G. Crist´obal, B. Escalante-Ram´ırez, M. J. Ruiz-Serrano, J. Pascau, and M. Desco, “Comparative Evaluation of Autofocus Algorithms for a Real-Time System for Automatic Detection of Mycobacterium Tuberculosis,” Cytometry Part A, vol. 81, no. 3, pp. 213–221, 2012.
- [3] C.-Y. Chen, R.-C. Hwang, and Y.-J. Chen, “A Passive Auto-Focus Camera Control System,” Applied Soft Computing, vol. 10, no. 1, pp. 296–303, 2010.
- [4] H. Mir, P. Xu, R. Chen, and P. van Beek, “An Autofocus Heuristic for Digital Cameras based on Supervised Machine Learning,” Journal of Heuristics, vol. 21, no. 5, pp. 599–616, 2015.
- [5] J. Li, “Autofocus Searching Algorithm Considering Human Visual System Limitations,” Optical Engineering, vol. 44, no. 11, p. 113201, 2005.
- [6] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, “A Brief Survey of Deep Reinforcement Learning,” arXiv preprint arXiv:1708.05866, 2017.
- [7] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing Atari with Deep Reinforcement Learning,” arXiv preprint arXiv:1312.5602, 2013.
- [8] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al., “Human-Level Control through Deep Reinforcement Learning,” Nature, vol. 518, no. 7540, p. 529, 2015.
- [9] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al., “Mastering the Game of Go without Human Knowledge,” Nature, vol. 550, no. 7676, p. 354, 2017.

Reference

- [10] J. He, R. Zhou, and Z. Hong, “Modified Fast Climbing Search Auto-Focus Algorithm with Adaptive Step Size Searching Technique for Digital Camera,” IEEE transactions on Consumer Electronics, vol. 49, no. 2, pp. 257–262, 2003.
- [11] N. Kehtarnavaz and H.-J. Oh, “Development and Real-Time Implementation of a Rule-based Auto-Focus Algorithm,” Real-Time Imaging, vol. 9, no. 3, pp. 197–203, 2003.
- [12] L. Metz, J. Ibarz, N. Jaitly, and J. Davidson, “Discrete Sequential Prediction of Continuous Actions for Deep RL,” arXiv preprint arXiv:1705.05035, 2017.
- [13] S. Levine, C. Finn, T. Darrell, and P. Abbeel, “End-to-End Training of Deep Visuomotor Policies,” The Journal of Machine Learning Research, vol. 17, no. 1, pp. 1334–1373, 2016.
- [14] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, “Trust Region Policy Optimization,” in International Conference on Machine Learning, pp. 1889–1897, 2015.
- [15] S. Gu, T. Lillicrap, I. Sutskever, and S. Levine, “Continuous Deep Q Learning with Model-based Acceleration,” in International Conference on Machine Learning, pp. 2829–2838, 2016.
- [16] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, “Continuous Control with Deep Reinforcement Learning,” arXiv preprint arXiv:1509.02971, 2015.
- [17] S. Gu, E. Holly, T. Lillicrap, and S. Levine, “Deep Reinforcement Learning for Robotic Manipulation with Asynchronous Off-Policy Updates,” in Robotics and Automation (ICRA), 2017 IEEE International Conference on, pp. 3389–3396, IEEE, 2017.
- [18] L. Ren, L. Wang, J. K. Mills, and D. Sun, “Vision-Based 2-D Automatic Micrograsping using Coarse-to-Fine Grasping Strategy,” IEEE Transactions on Industrial Electronics, vol. 55, no. 9, pp. 3324–3331, 2008.

Reference

- [19] S. J. Ralis, B. Vikramaditya, and B. J. Nelson, “Micropositioning of a Weakly Calibrated Microassembly System using Coarse-to-Fine Visual Servoing Strategies,” IEEE Transactions on Electronics Packaging Manufacturing, vol. 23, no. 2, pp. 123–131, 2000.
- [20] M. Riedmiller, “Neural Fitted Q Iteration–First Experiences with a Data Efficient Neural Reinforcement Learning Method,” in European Conference on Machine Learning, pp. 317–328, Springer, 2005.
- [21] S. Levine, P. Pastor, A. Krizhevsky, and D. Quillen, “Learning Hand-Eye Coordination for Robotic Grasping with Large-Scale Data Collection,” in International Symposium on Experimental Robotics, pp. 173–184, Springer, 2016.