End-to-End Training of Deep Visuomotor Policies

Sergey Levine*, Chelsea Finn*, Trevor Darrell, Pieter Abbeel

***Abstract*—Policy search methods can allow robots to automatically learn control policies for a wide range of tasks. However, practical applications of policy search tend to require the policy to be supported by hand-engineered components for perception, state estimation, and low-level control. In this paper, we aim to answer the following question: does training the perception and control systems jointly end-to-end provide for better performance than training each component separately? To answer this question, we develop a method that can be used to train policies that map raw observations, that consist of joint angles and image pixels, directly to torques at the robot's motors. The policies are represented by deep convolutional neural networks (CNNs) with 92,000 parameters, and are trained by using guided policy search, which transforms the policy search task into a supervised learning problem. We show that this method can learn manipulation tasks that require close coordination between vision and control, including inserting a block into a shape sorting cube, screwing on a bottle cap, fitting the claw of a toy hammer under a nail with various grasps, and placing a coat hanger on a clothes rack.**

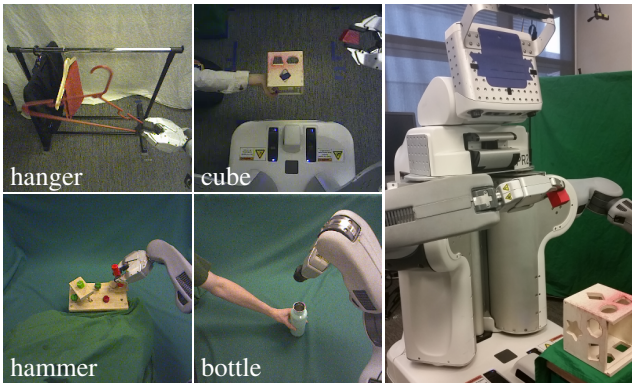


Fig. 1: Our method learns visuomotor policies that directly use camera image observations (left) to set motor torques on a PR2 robot (right).

I. Introduction

Policy search methods can allow robots to autonomously learn a wide variety of behaviors. However, policies learned using such methods often rely on hand-engineered components for perception and low-level control. For example, a policy for object manipulation might specify motions in task-space, using hand-designed PD controllers to execute the desired motion and relying on an existing vision system to localize objects in the scene [8]. The vision system in particular can be complex and prone to errors, and its performance is typically not improved during policy training, nor adapted to the goal of the task.

We propose a method for learning policies that directly map camera images and joint angles to motor torques. The policies are trained end-to-end using real-world experience, optimizing both the control and vision components on the same measure of task performance. This allows the policy to learn goal-driven perception, which avoids the mistakes that are most costly for task performance. Learning perception and control in a general and flexible way requires a large, expressive model. We use convolutional neural networks (CNNs), which have 92,000 parameters and 7 layers. Deep CNN models have achieved state of the art results on supervised vision tasks [3, 9], but sensorimotor deep learning remains a challenging prospect. The policies are extremely high dimensional, and the control task is partially observed, since part of the state must be inferred from images.

To address these challenges, we extend the framework of guided policy search to sensorimotor deep learning. Guided policy search decomposes policy search into two phases: a trajectory optimization phase that determines how to solve the task in a few specific conditions, and a supervised learning phase that trains the policy from these successful executions with supervised learning [6]. Since the CNN policy is trained with supervised learning, we can use the tools developed in the deep learning community to make this phase simple and efficient. We handle the partial observability of visuomotor control by optimizing the trajectories with full state information, while providing only partial observations (consisting of images and robot configurations) to the policy. The trajectories are optimized under unknown dynamics, using real-world experience and minimal prior knowledge.

The main contribution of our work is a method for end-to-end training of deep visuomotor policies for robotic manipulation. This includes a partially observed guided policy search algorithm that can train high-dimensional policies for tasks where part of the state must be determined from camera images, as well as a novel CNN architecture designed for robotic control, shown in Figure 1. Our results demonstrate improvements in consistency and generalization from training visuomotor policies end-to-end, when compared to the more standard approach of training the vision and control components separately. A complete description of our work can be found in our recent technical report [5], and videos of the learned policies can be found on the project website[1].

II. Visuomotor Policy Architecture

The aim of our method is to learn a policy $\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ that specifies a distribution over actions $\mathbf{u}_t$ conditioned on the observation $\mathbf{o}_t$, which includes a camera image and the configuration of the robot, which consists of the joint angles,

[1]The video can be viewed at http://sites.google.com/site/visuomotorpolicy

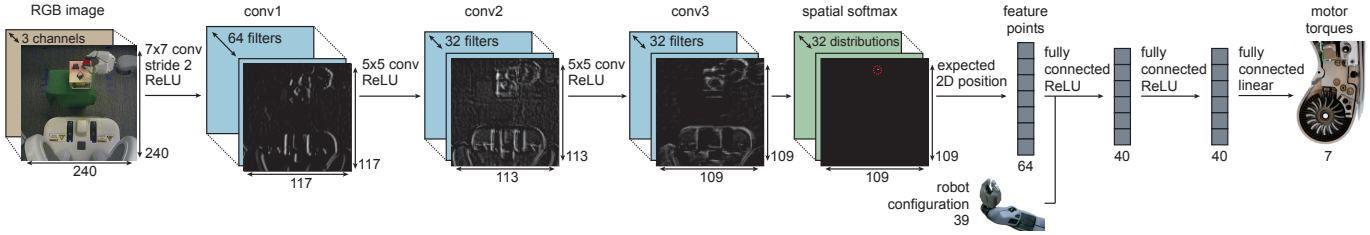


Fig. 2: Visuomotor policy architecture. The network contains three convolutional layers, followed by a spatial softmax and an expected position layer that converts pixel-wise features to feature points, which are better suited for spatial computations. The points are concatenated with the robot configuration, then passed through three fully connected layers to produce the torques.

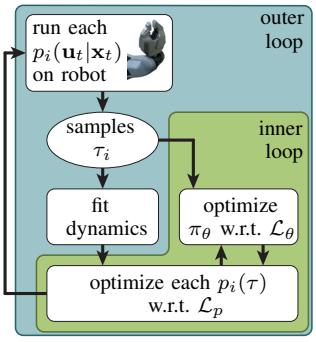
end-effector position, and their velocities. Our policies run on a PR2 robot at 20 Hz. The policy parameters θ are optimized to minimize a cost function $\ell(\mathbf{x}_t, \mathbf{u}_t)$ over the course of a fixed-length episode. The actions $\mathbf{u}_t$ are the motor torques, and the state $\mathbf{x}_t$ includes the known robot configuration, as well as (for example) the target position for an object placement task. The latter information is not observed directly by the policy, and must be inferred from the camera image. We represent $\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ as a Gaussian, with the mean given by a nonlinear function approximator. Since this function approximator needs to operate directly on raw images, we use convolutional neural networks (CNNs). The architecture of our CNN is shown in Figure 2. This network has 7 layers and around 92,000 parameters, which presents a tremendous challenge for standard policy search methods [1].

CNNs built for spatial tasks such as human pose estimation often rely on the availability of location labels in image-space, such as hand-labeled keypoints [9]. We propose a novel CNN architecture capable of estimating spatial information from an image without direct supervision in image space. The core component of our network architecture, shown in Figure 2 is a spatial feature point transformation that consists of a softmax followed by an expectation operator. Intuitively, the role of this transformation is to find the point of maximal activation in each channel of the last convolutional layer, creating a kind of soft arg-max. Formally, the activations in each of the 32 response maps in the last convolutional layer are passed through a spatial softmax function of the form $s_{cij} = e^{a_{cij}} / \sum_{i'j'} e^{a_{ci'j'}}$. Each output channel of the softmax is a probability distribution over the location of a feature in the image. To convert from this distribution to a coordinate representation, the network calculates the expected image position of each feature, yielding a 2D coordinate for each channel. This corresponds to a fully connected layer with weights corresponding to image-space positions of each point in the response map. The resulting spatial feature points are concatenated with the robot's configuration and fed through two fully connected layers, each with 40 rectified units, followed by linear connections to the torques. The full visuomotor policy contains about 92,000 parameters, of which 86,000 are in the convolutional layers.

The spatial softmax and the expected position computation serve to convert pixel-wise representations in the convolutional layers to spatial coordinate representations, which can be manipulated by the fully connected layers into 3D positions or motor torques. The softmax also provides lateral inhibition, which suppresses low, erroneous activations. Our experiments show that this network can learn useful visual features using only 3D positional information provided by the robot and no camera calibration. Furthermore, by training our network with guided policy search, it can acquire *task-specific* visual features that improve policy performance.

III. Visuomotor Policy Training

The high dimensionality of our CNN policies makes them extremely difficult to optimize with standard reinforcement learning methods [1]. Guided policy search [4] transforms policy search into a supervised learning problem, which can be used to optimize much higher dimensional policies. The training set for supervised learning is generated by simple trajectory-centric algorithms. The trajectory phase produces Gaussian trajectory distributions $p_i(\tau)$, which correspond to a mean trajectory with linear feedback. Each $p_i(\tau)$ succeeds from a specific initial state. For example, in the task of placing a cap on a bottle, these initial states might be different bottle positions. By training on multiple trajectories for multiple bottle positions, the final CNN policy can succeed from all initial states, and can generalize to other states from the same distribution.



We present a partially observed guided policy search method that uses BADMM to iteratively enforce agreement between the policy $\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ and the trajectory distributions $p_i(\tau)$. A diagram of this method is shown on the right. In the outer loop, we draw a sample for each initial state on the real system. The samples are used to fit the dynamics for trajectory optimization, and serve as training data for the policy. The inner loop alternates between optimizing each $p_i(\tau)$ and optimizing the policy. Unlike prior guided policy search methods, the policy is trained on observations $\mathbf{o}_t$, allowing the method to handle partial observability, while the

trajectories are optimized on the full state $\mathbf{x}_t$. For example, if the unobserved part of $\mathbf{x}_t$ is the position of a target object, such as the bottle, we can hold this object in the robot's left gripper, while the right arm performs the task. This type of instrumented training is a natural fit for many robotic tasks, where training is performed in a controlled environment, but the final policy must be able to succeed "in the wild."

IV. Experimental Results

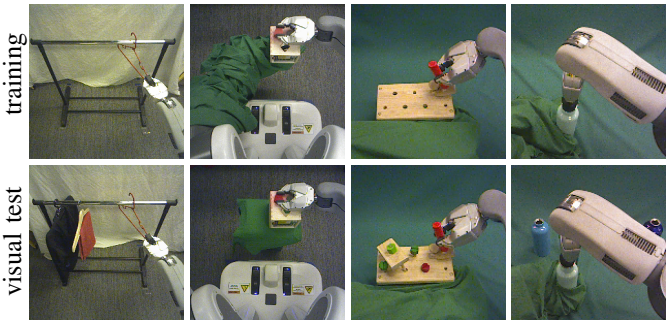
We evaluated our method by training policies on a PR2 robot for hanging a coat hanger on a clothes rack, inserting a block into a shape sorting cube, fitting the claw of a toy hammer under a nail with various grasps, and screwing on a bottle cap. The cost function for these tasks encourages low distance between three points on the end-effector and corresponding target points, low torques, and, for the bottle task, spinning the wrist. The equations for these cost functions follow prior work [6]. The tasks are illustrated in Figure 1. Each task involved variation of about 10-20 cm in each direction in the position of the target object (the rack, shape sorting cube, nail, and bottle). In addition, the coat hanger and hammer tasks were trained with two and three grasps, respectively. All tasks used the same policy architecture.

We evaluated the visuomotor policies in three conditions: (1) the training target positions and grasps, (2) new target positions not seen during training and, for the hammer, new grasps (spatial test), and (3) training positions with visual distractors (visual test). A selection of these experiments can be viewed in the videos available online[2]. For the visual test, the shape sorting cube was placed on a table, the coat hanger was placed on a rack with clothes, and the bottle and hammer tasks were performed in the presence of clutter. Illustrations of this test are shown in Figure 3.

The success rates for each test are shown in Table I. We compared to two baselines, both of which train the vision layers in advance for pose prediction, instead of training the entire policy end-to-end. The features baseline discards the last layer of the pose predictor and uses the feature points, resulting in the same architecture as our policy, while the prediction baseline feeds the predicted pose into the control layers.

The pose prediction baseline is analogous to a standard modular approach to policy learning, where the vision system is first trained to localize the target, and the policy is trained on top of it. This variant achieves poor performance, because although the pose is accurate to about 1 cm, this is insufficient for such precise tasks. As shown in the video, the shape sorting cube and bottle cap insertions have tolerances of just a few millimeters. Such accuracy is difficult to achieve even with calibrated cameras and checkerboards. Indeed, prior work has reported that the PR2 can maintain a camera to end effector accuracy of about 2 cm during open loop motion [7]. This suggests that the failure of this baseline is not atypical, and that our visuomotor policies are learning visual features and control strategies that improve the robot's accuracy.

[2]The video can be viewed at http://sites.google.com/site/visuomotorpolicy



(a) hanger (b) cube (c) hammer (d) bottle

Fig. 3: Training and visual test scenes as seen by the policy at the ends of successful episodes. The hammer and bottle images were cropped for visualization only.

When provided with pose estimation features, the policy has more freedom in how it uses the visual information, and achieves somewhat higher success rates. However, full end-to-end training performs significantly better, achieving high accuracy even on the challenging bottle task, and successfully adapting to the variety of grasps in the hammer task. This suggests that, although the vision layer pre-training is clearly beneficial for reducing computation time, it is not sufficient by itself for discovering good features for visuomotor policies.

coat hanger	training (18)	spatial test (24)	visual test (18)
end-to-end training	**100%**	**100%**	**100%**
pose features	88.9%	87.5%	83.3%
pose prediction	55.6%	58.3%	66.7%
shape sorting cube	training (27)	spatial test (36)	visual test (40)
end-to-end training	**96.3%**	**91.7%**	**87.5%**
pose features	70.4%	83.3%	40%
pose prediction	0%	0%	n/a
toy claw hammer	training (45)	spatial test (60)	visual test (60)
end-to-end training	**91.1%**	**86.7%**	**78.3%**
pose features	62.2%	75.0%	53.3%
pose prediction	8.9%	18.3%	n/a
bottle cap	training (27)	spatial test (12)	visual test (40)
end-to-end training	**88.9%**	**83.3%**	**62.5%**
pose features	55.6%	58.3%	27.5%

TABLE I: Success rates on training positions, on novel test positions, and in the presence of visual distractors. The number of trials per test is shown in parentheses.

The policies exhibit moderate tolerance to distractors that are visually separated from the target object. However, as expected, they tend to perform poorly under drastic changes to the backdrop, or when the distractors are adjacent to or occluding the manipulated objects, as shown in the videos available online. In future work, this could be mitigated by varying the scene at training time, or by artificially augmenting the image samples in the training set with synthetic transformations.

V. Discussion and Future Work

We presented a method for learning robotic control policies that use raw camera input. The policies are represented by a novel convolutional neural network architecture, and can be trained end-to-end using our partially observed guided

policy search algorithm, which decomposes the policy search problem in a trajectory optimization phase that uses full state information and a supervised learning phase that only uses the camera observations. This decomposition allows us to leverage state-of-the-art tools from supervised learning and optimize high-dimensional CNN policies. Our experimental results show that end-to-end training produces significant improvements in policy performance compared to using fixed vision layers trained for pose prediction on a real robotic manipulator.

Although we demonstrate moderate generalization over variations in the scene, our current method does not generalize to dramatically different settings, especially when visual distractors occlude the manipulated object. The success of CNNs on exceedingly challenging vision tasks suggests that this class of models is capable of learning invariance to irrelevant distractor features [2, 3, 9], and in principle this issue can be addressed by training the policy in a variety of environments, though this poses certain logistical challenges. More practical alternatives that could be explored in future work include simultaneously training the policy on multiple robots in a different environment, developing more sophisticated regularization and pre-training, and introducing synthetic data augmentation. However, even without these improvements, our method has numerous applications in, for example, industrial settings where the robot must repeatedly and efficiently perform a task that requires visual feedback under moderate variation in background and clutter conditions.

In future work, we hope to explore recurrent policies that can deal with extensive occlusions by keeping a memory of past observations. We also hope to extend our method to a variety of other rich sensory modalities, including haptic input from pressure sensors and auditory input. We expect that end-to-end training will become increasingly important with more varied sensory modalities, where it is much less apparent how to manually engineer the appropriate perception modules.

References

[1] M. Deisenroth, G. Neumann, and J. Peters. A survey on policy search for robotics. *Foundations and Trends in Robotics*, 2(1-2):1–142, 2013.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[3] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*. 2012.

[4] S. Levine and V. Koltun. Guided policy search. In *International Conference on Machine Learning (ICML)*, 2013.

[5] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *arXiv preprint arXiv:1504.00702*, 2015.

[6] S. Levine, N. Wagener, and P. Abbeel. Learning contact-rich manipulation skills with guided policy search. In *International Conference on Robotics and Automation (ICRA)*, 2015.

[7] W. Meeussen, M. Wise, S. Glaser, S. Chitta, C. McGann, P. Mihelich, E. Marder-Eppstein, M. Muja, Victor Eruhimov, T. Foote, J. Hsu, R.B. Rusu, B. Marthi, G. Bradski, K. Konolige, B. Gerkey, and E. Berger. Autonomous door opening and plugging in with a personal robot. In *International Conference on Robotics and Automation (ICRA)*, 2010.

[8] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal. Learning and generalization of motor skills by learning from demonstration. In *International Conference on Robotics and Automation (ICRA)*, 2009.

[9] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.