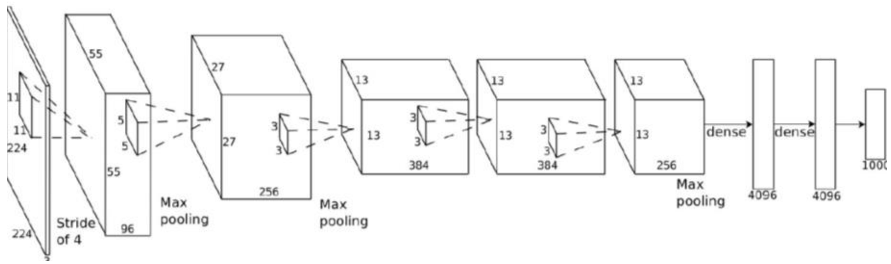
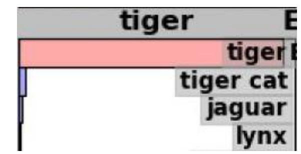
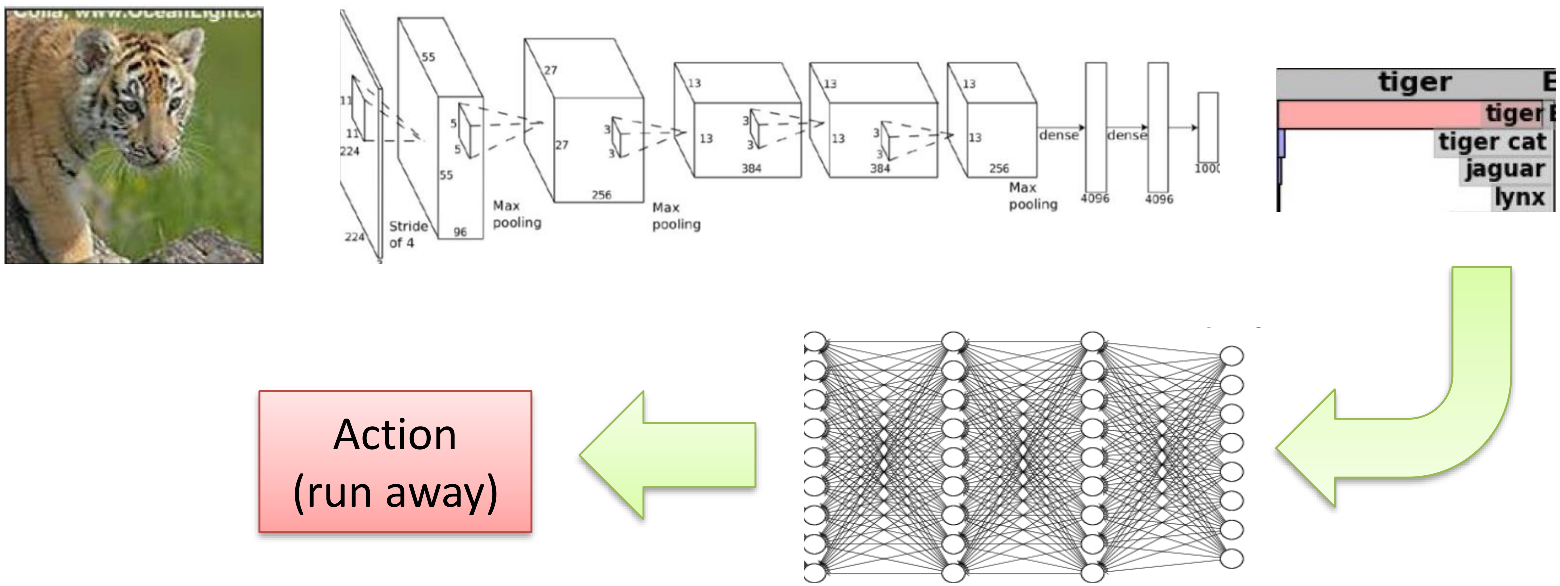
Deep Robotic Learning

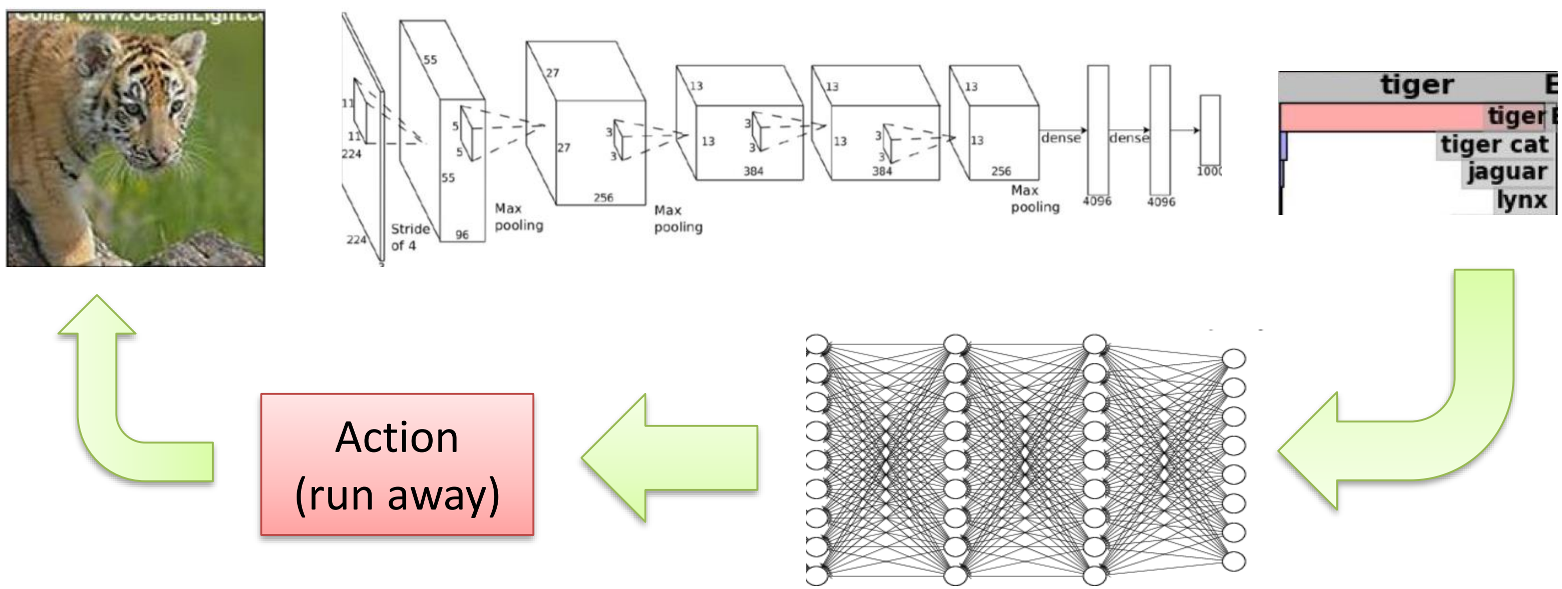
Sergey Levine

University of Washington

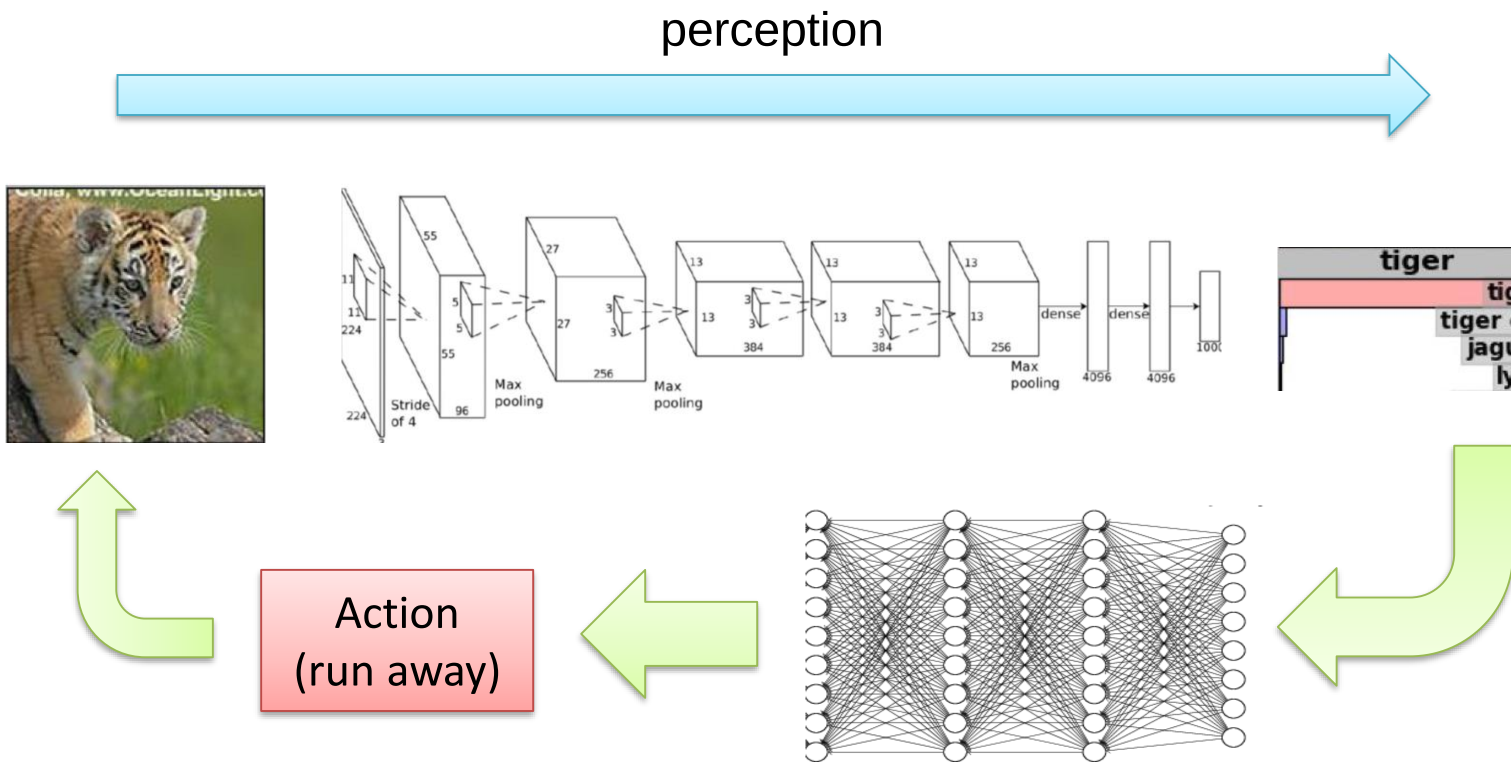

11
11
224
224
Stride
of 4
55
55
96
5
5
Max
pooling
27
27
256
Max
pooling
3
3
13
13
384
3
3
13
13
384
3
3
13
13
256
Max
pooling
dense
dense
4096
4096

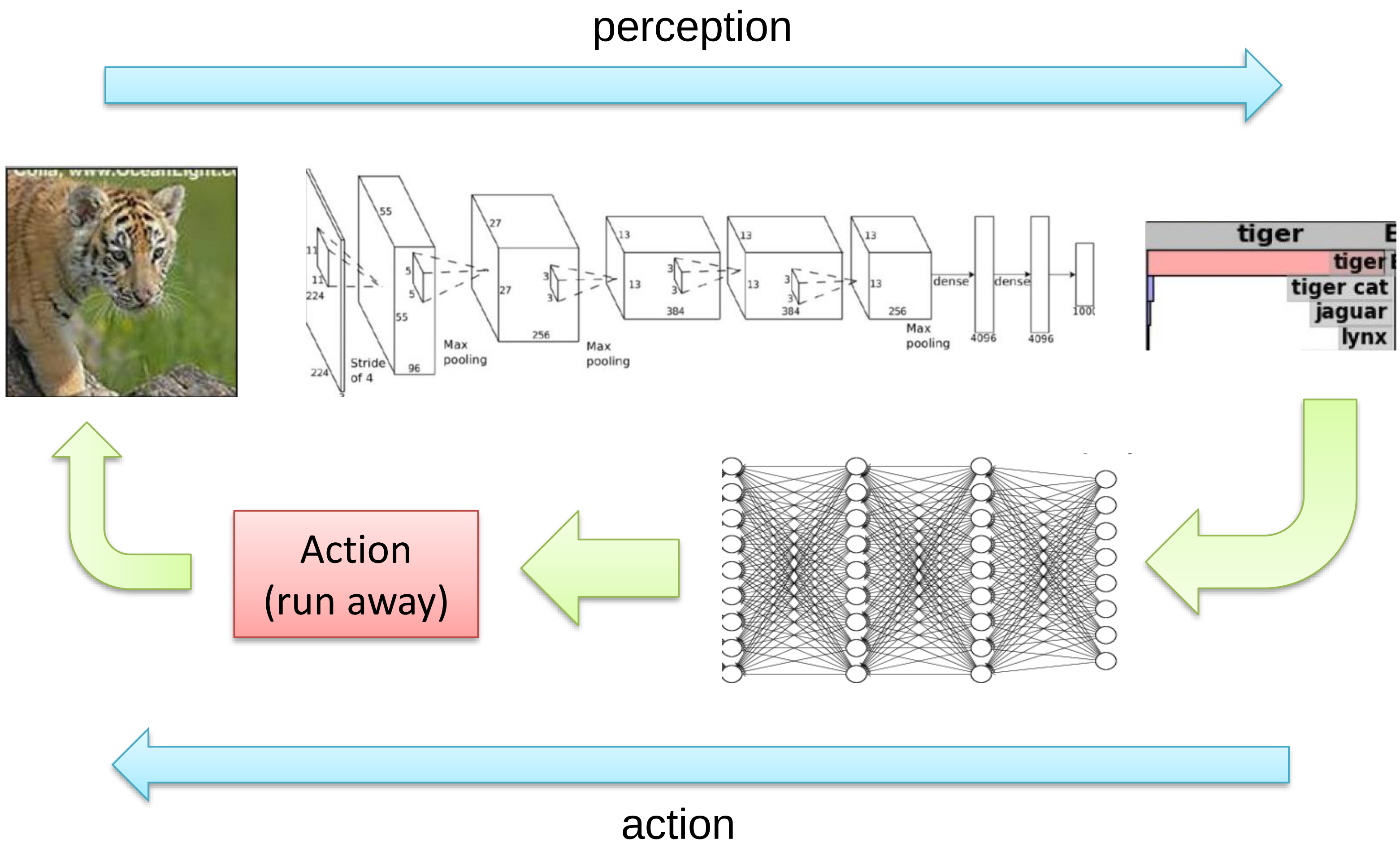

tiger
tiger
tiger cat
jaguar
lynx


224
224
Stride of 4
55
55
96
Max pooling
27
27
256
Max pooling
13
13
384
13
13
384
13
13
256
Max pooling
dense
dense
4096
4096
tiger
tiger
tiger cat
jaguar
lynx
Action
(run away)



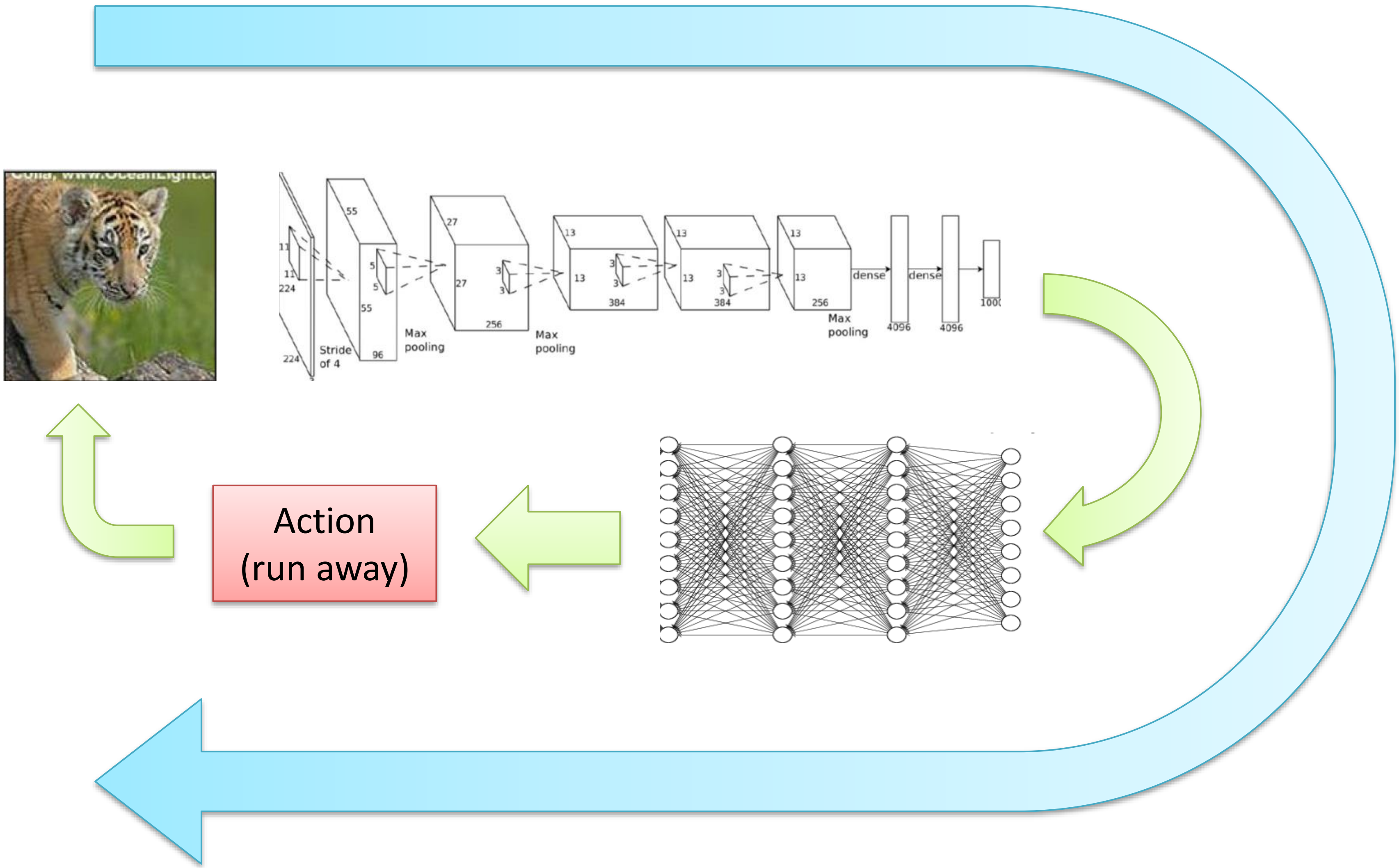
224
224
Stride of 4
55
55
96
Max pooling
27
27
256
Max pooling
13
13
384
13
13
384
13
13
256
Max pooling
dense
dense
4096
4096
tiger
tiger
tiger cat
jaguar
lynx
Action
(run away)


perception
Stride of 4
Max pooling
Max pooling
Max pooling
dense
dense
4096
4096
tiger
tiger
tiger cat
jaguar
lynx
Action
(run away)



perception
Stride of 4
Max pooling
Max pooling
Max pooling
dense
dense
tiger
tiger
tiger cat
jaguar
lynx
Action
(run away)
action

sensorimotor loop

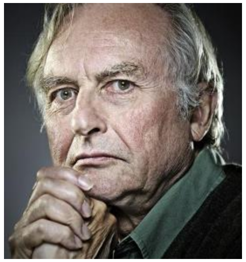


"When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on."

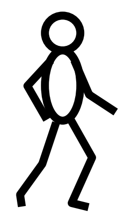
-- Richard Dawkins

"When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on."

-- Richard Dawkins



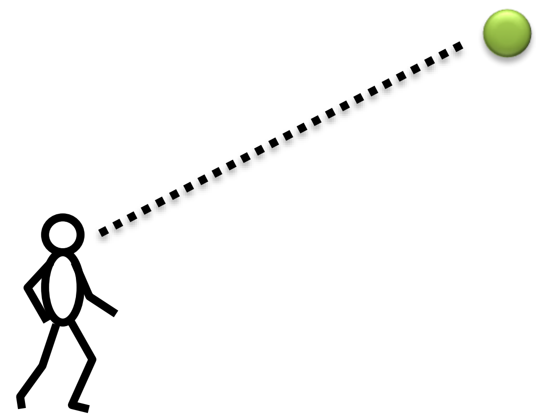
McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543



"When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on."

-- Richard Dawkins

McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543



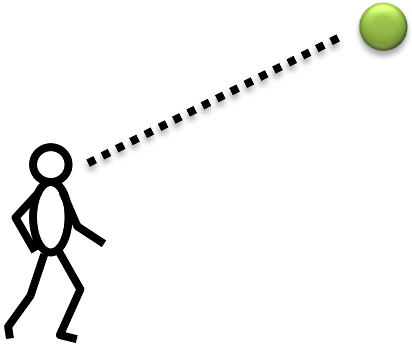
“When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on.”

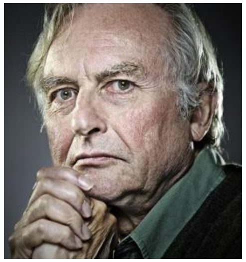
-- Richard Dawkins

McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543

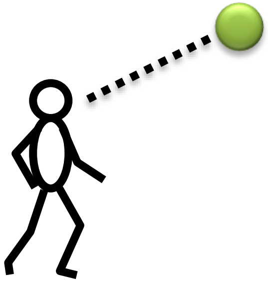
“When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on.”

-- Richard Dawkins



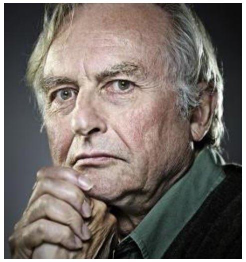


McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543



"When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on."

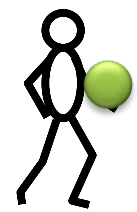
-- Richard Dawkins



McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543

"When a man throws a ball high in the air and catches it again, he behaves as if he had solved a set of differential equations in predicting the trajectory of the ball … at some subconscious level, something functionally equivalent to the mathematical calculations is going on."

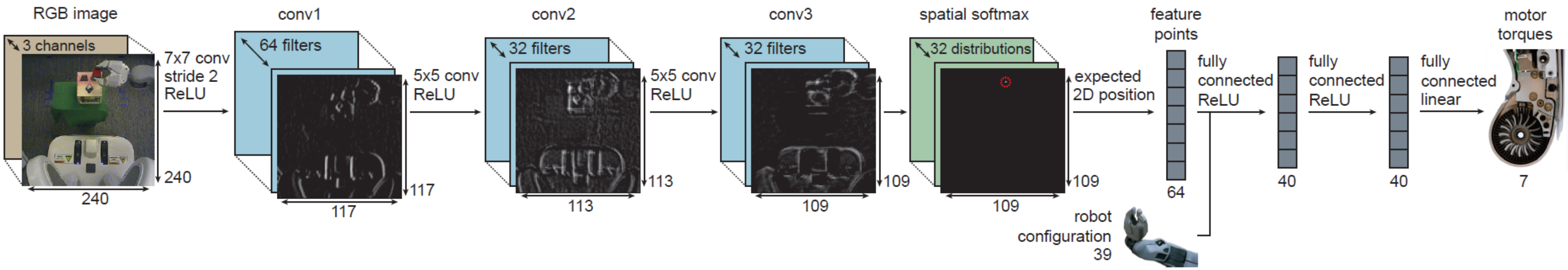
-- Richard Dawkins

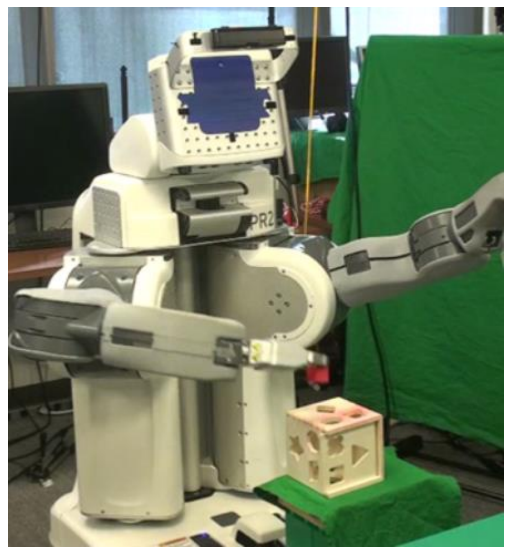


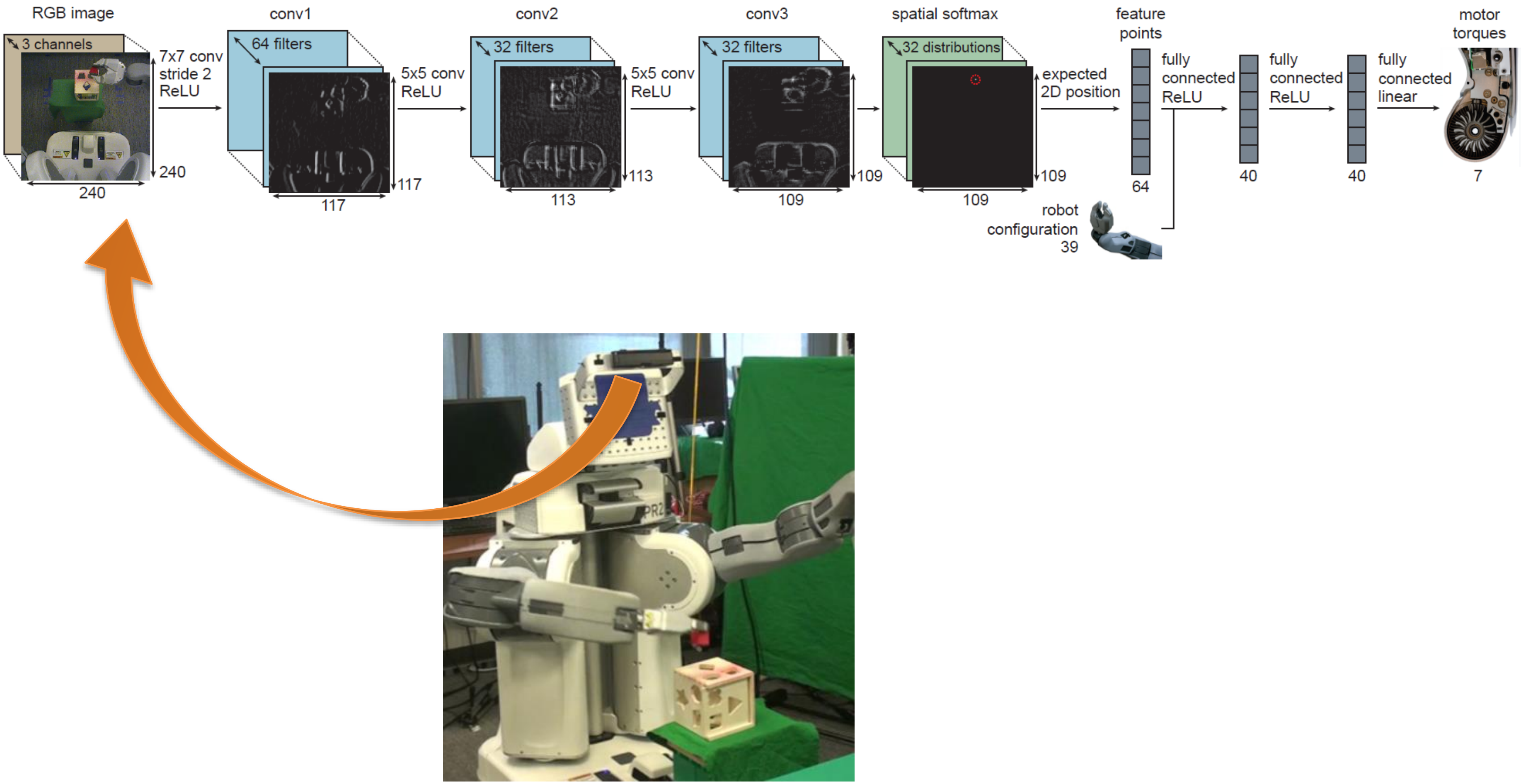
McLeod & Dienes. Do fielders know where to go to catch the ball or only how to get there? Journal of Experimental Psychology 1996, Vol. 22, No. 3, 531-543

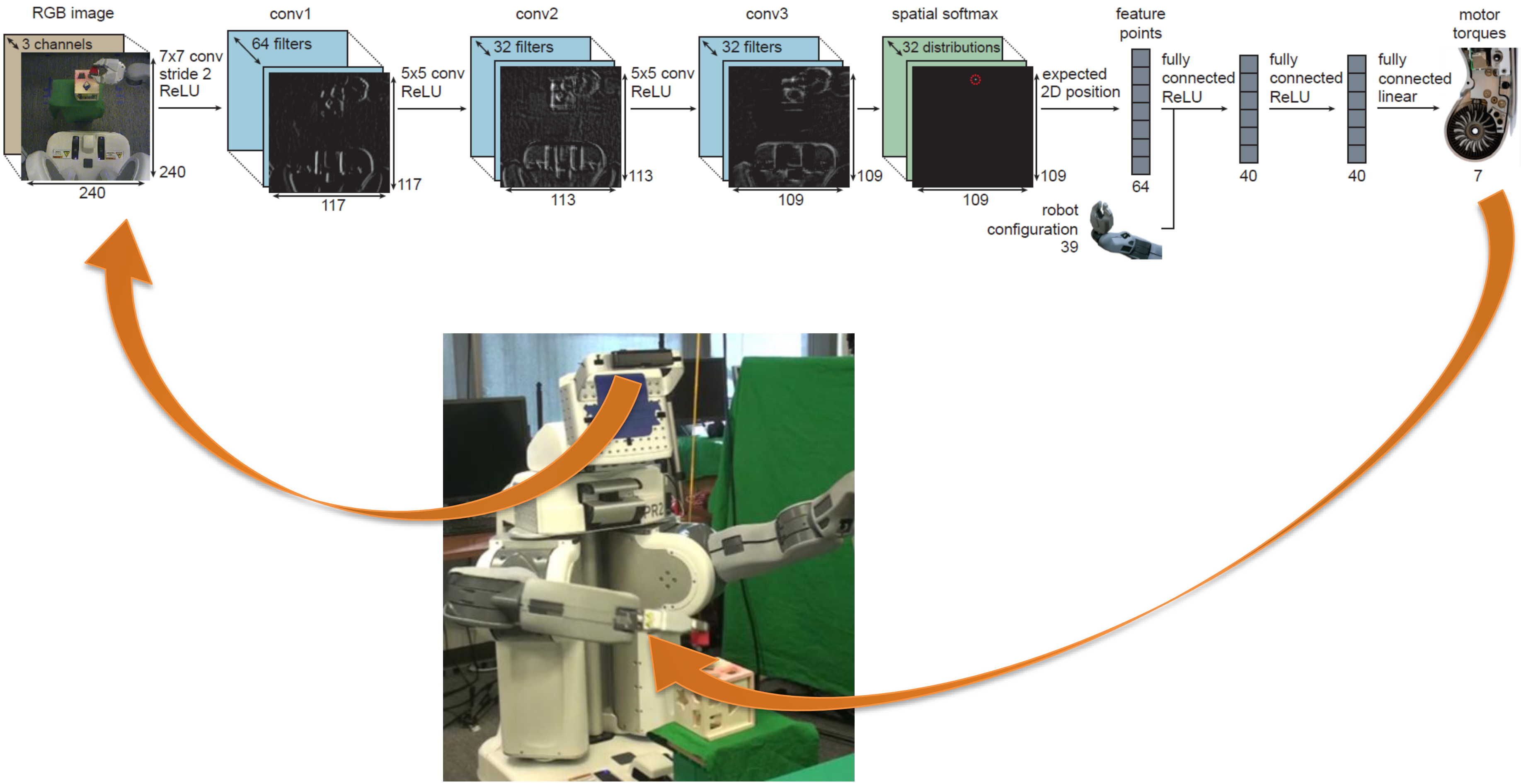
KAIST’s DRC-HUBO opening a door

DARPA Robotics Challenge 2015

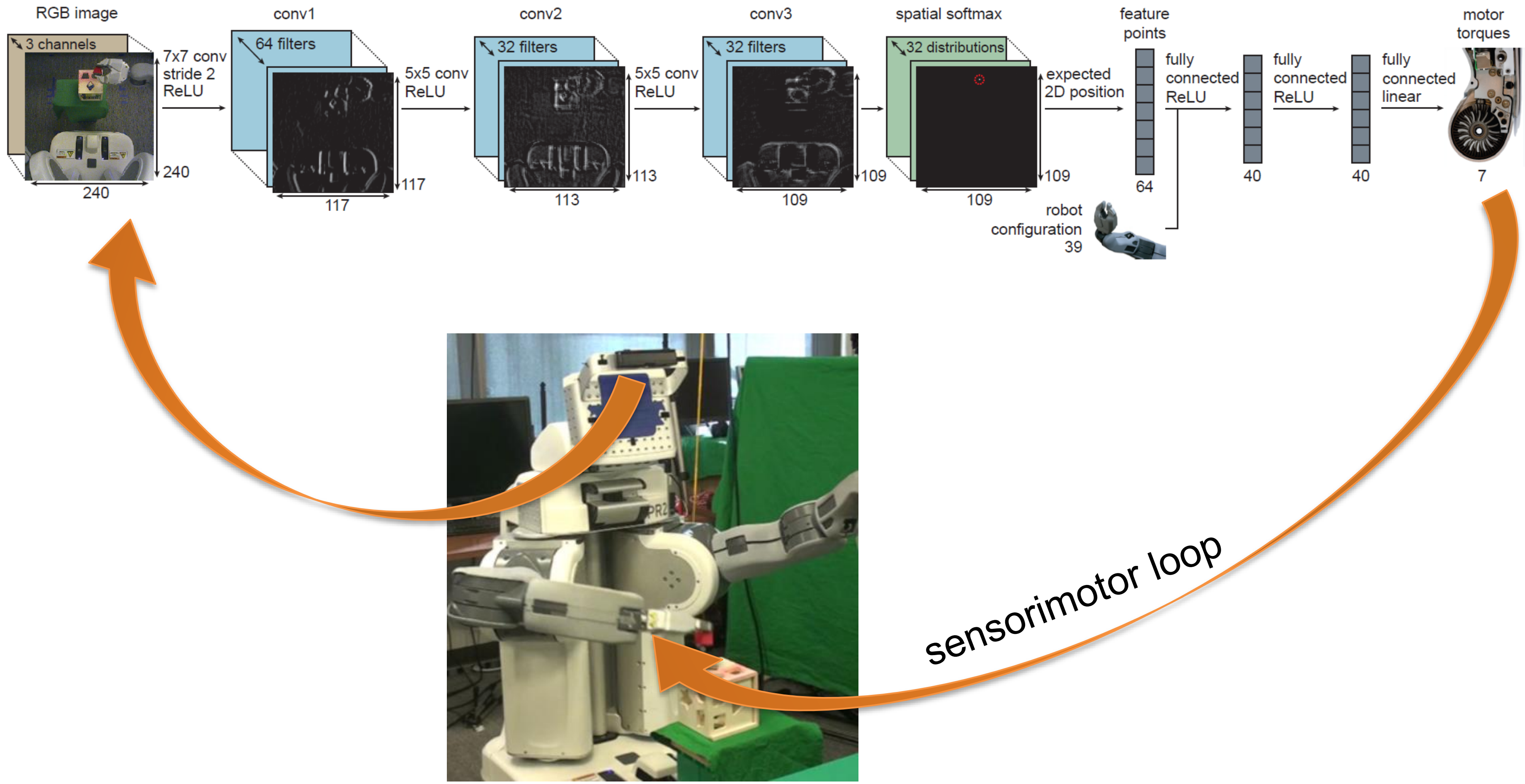

RGB image
3 channels
7x7 conv
stride 2
ReLU
240
240
conv1
64 filters
117
117
5x5 conv
ReLU
conv2
32 filters
113
113
5x5 conv
ReLU
conv3
32 filters
109
109
spatial softmax
32 distributions
109
109
expected
2D position
feature
points
64
robot
configuration
39
fully
connected
ReLU
40
fully
connected
ReLU
40
fully
connected
linear
motor
torques
7

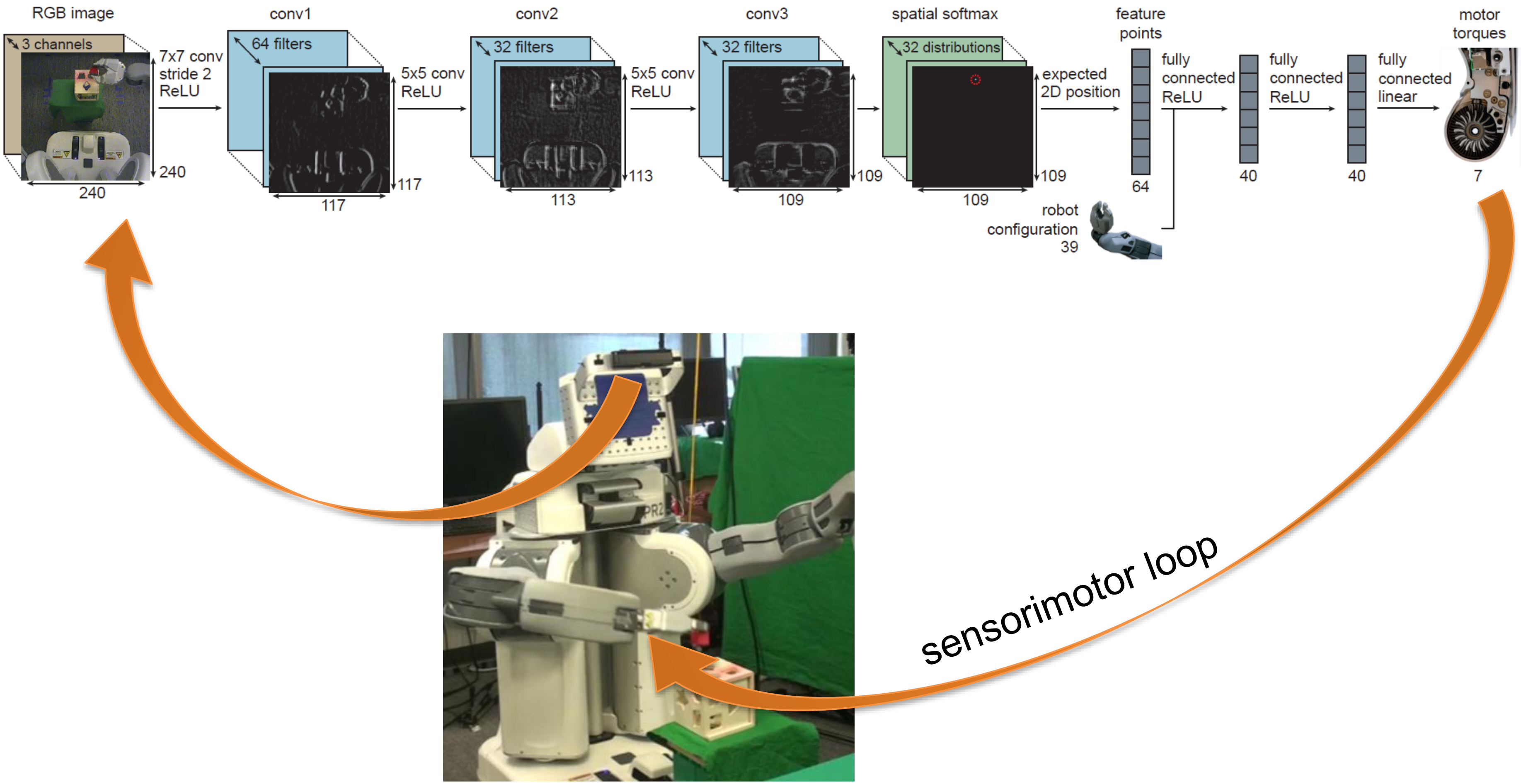



RGB image
3 channels
240
240
7x7 conv
stride 2
ReLU
conv1
64 filters
117
117
5x5 conv
ReLU
conv2
32 filters
113
113
5x5 conv
ReLU
conv3
32 filters
109
109
spatial softmax
32 distributions
109
109
expected
2D position
feature
points
64
robot
configuration
39
fully
connected
ReLU
40
fully
connected
ReLU
40
fully
connected
linear
motor
torques
7
PR2

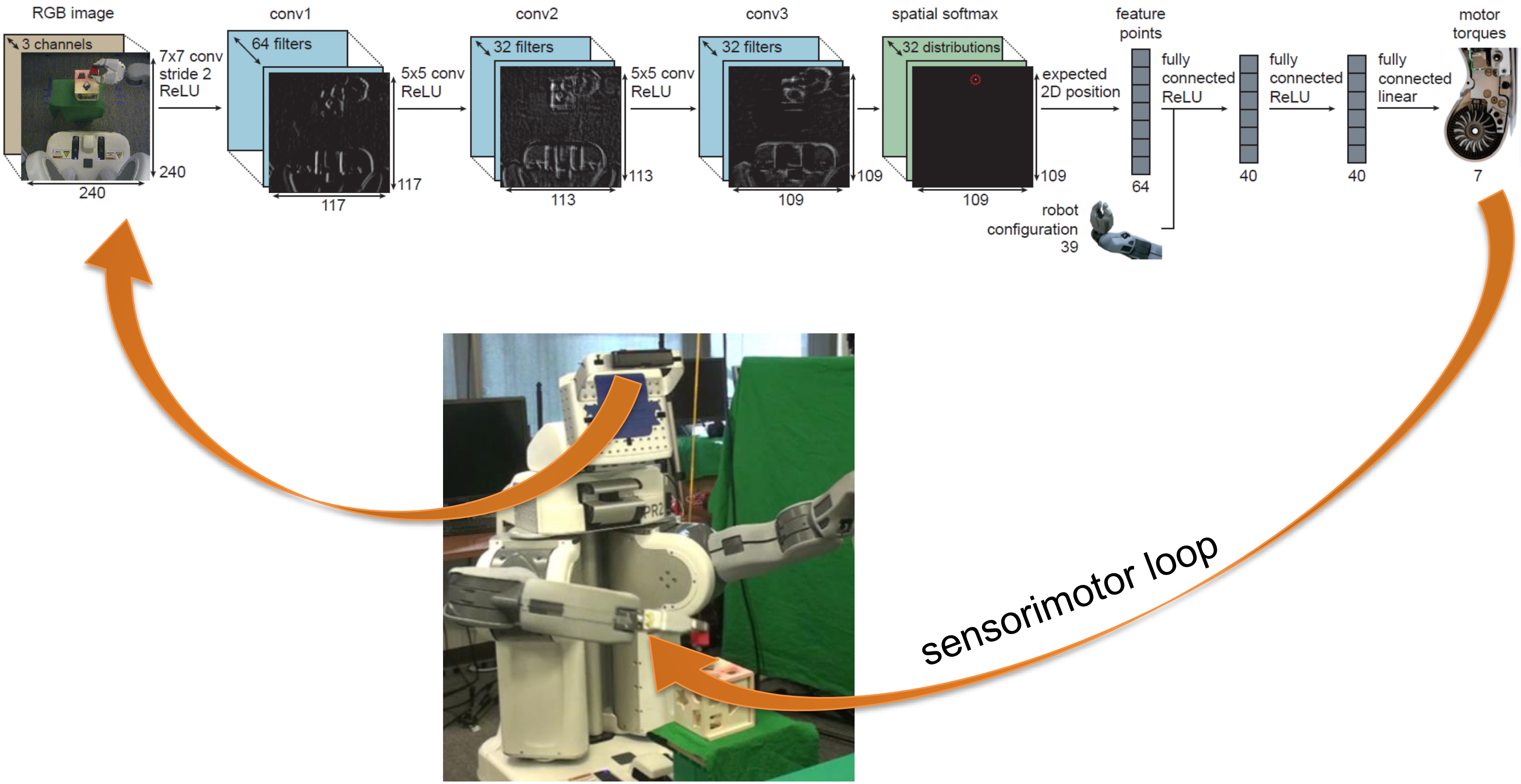


RGB image
3 channels
240
240
7x7 conv
stride 2
ReLU
conv1
64 filters
117
117
5x5 conv
ReLU
conv2
32 filters
113
113
5x5 conv
ReLU
conv3
32 filters
109
109
spatial softmax
32 distributions
109
109
expected
2D position
feature
points
64
robot
configuration
39
fully
connected
ReLU
40
fully
connected
ReLU
40
fully
connected
linear
motor
torques
7


RGB image
3 channels
240
240
7x7 conv
stride 2
ReLU
conv1
64 filters
117
117
5x5 conv
ReLU
conv2
32 filters
113
113
5x5 conv
ReLU
conv3
32 filters
109
109
spatial softmax
32 distributions
109
109
expected
2D position
feature
points
64
fully
connected
ReLU
40
fully
connected
ReLU
40
fully
connected
linear
motor
torques
7
robot
configuration
39
sensorimotor loop



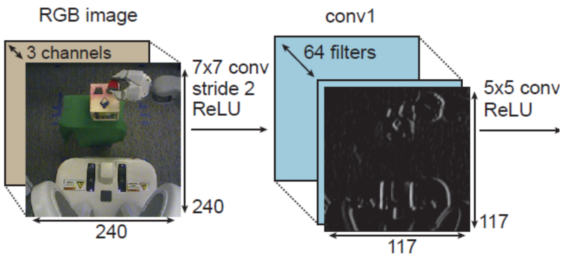
no direct supervision



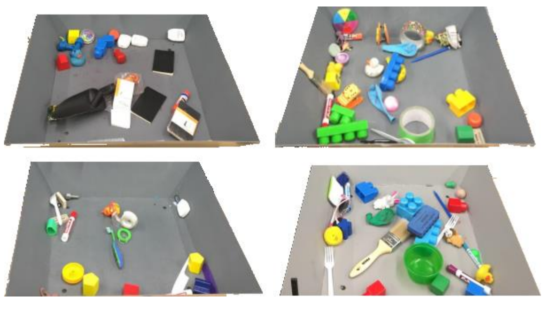
no direct supervision

actions have consequences

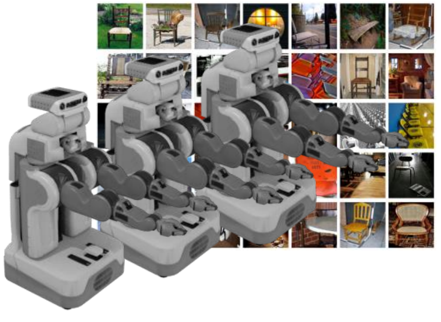
Overview



Training visuomotor policies

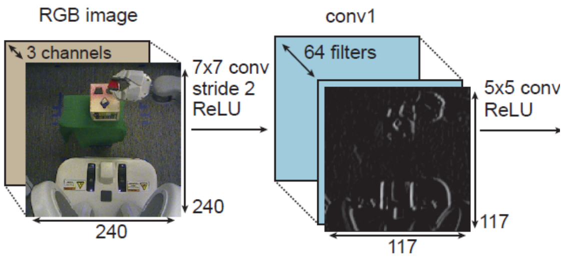


Deep robotic learning at scale



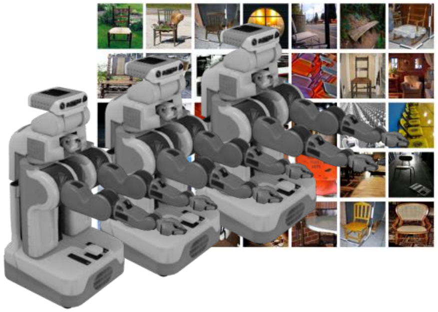
Future directions

Overview



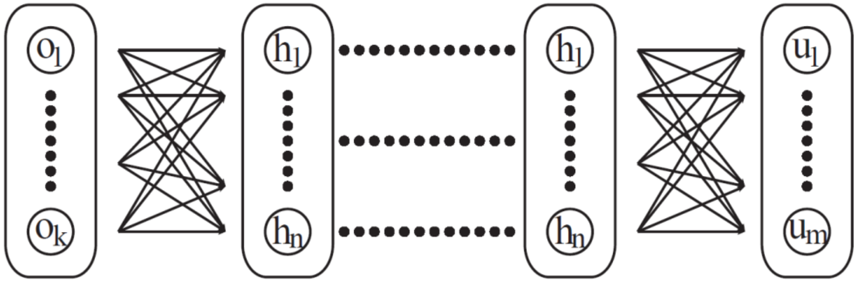
Training visuomotor policies

Deep robotic learning at scale

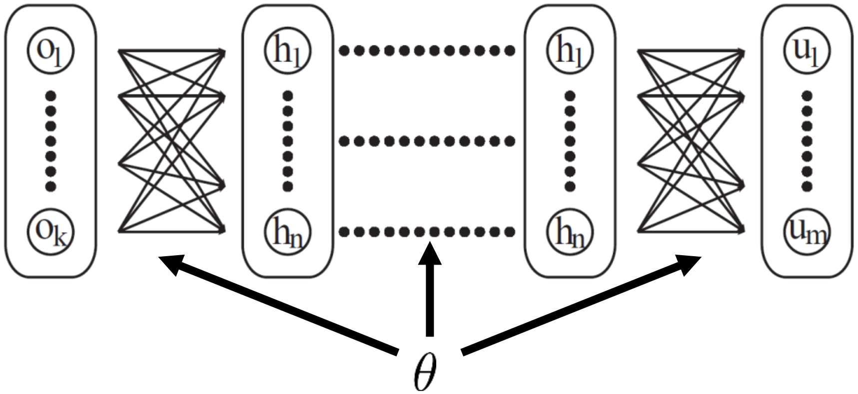


Future directions

general-purpose neural network policy



general-purpose neural network policy

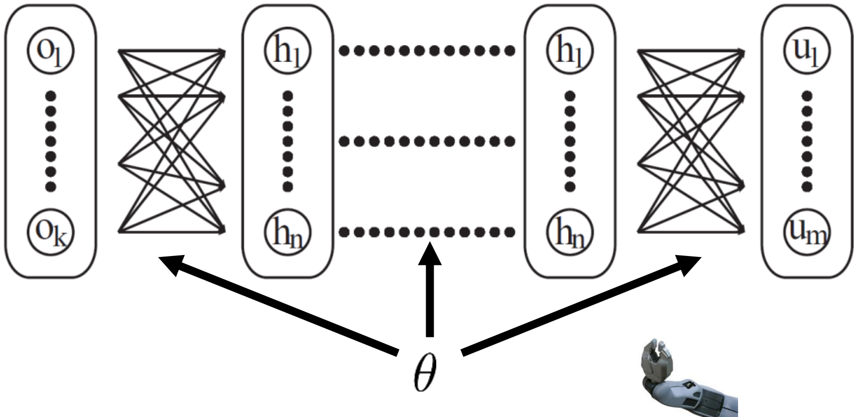


$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^T c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t\text{)}$$

general-purpose neural network policy

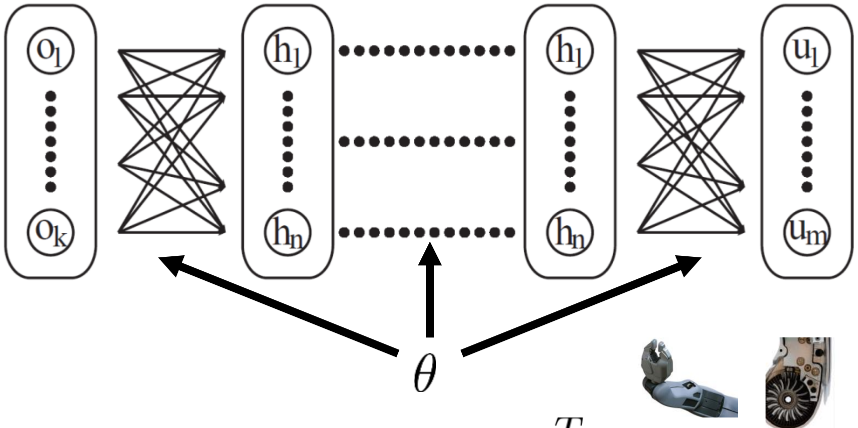


$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t)$$

general-purpose neural network policy

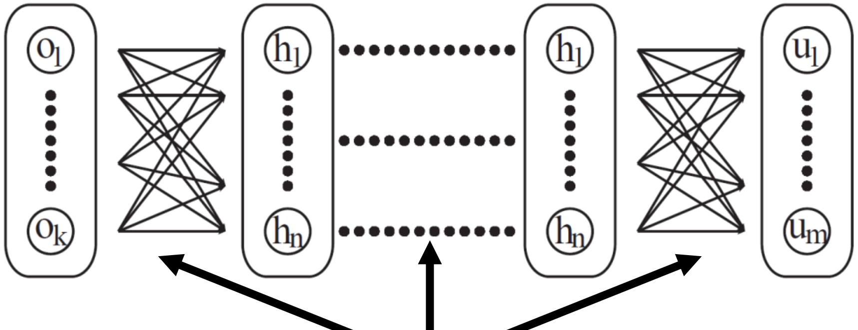


$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ – control policy

$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)

general-purpose neural network policy

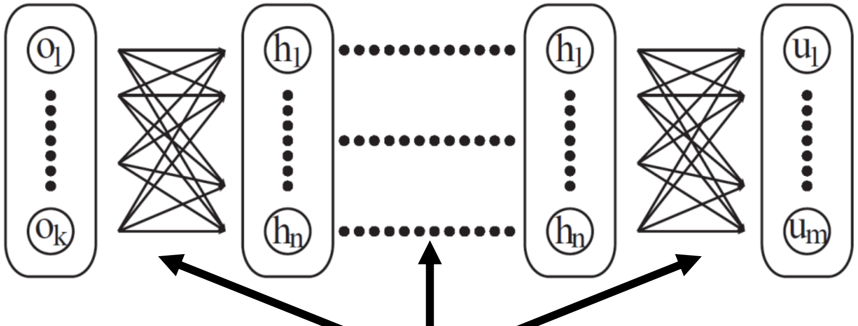


$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

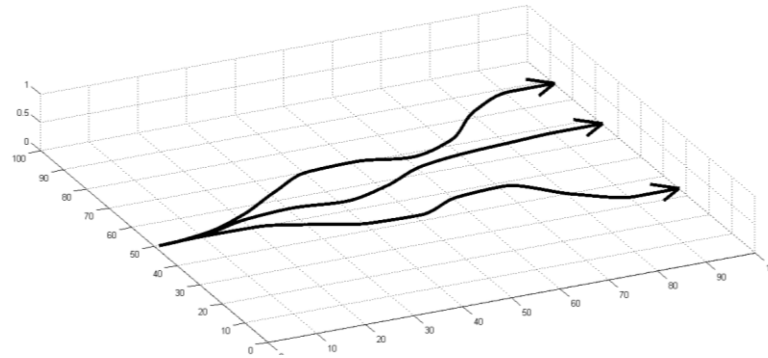
$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ – control policy

$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)

general-purpose neural network policy



θ

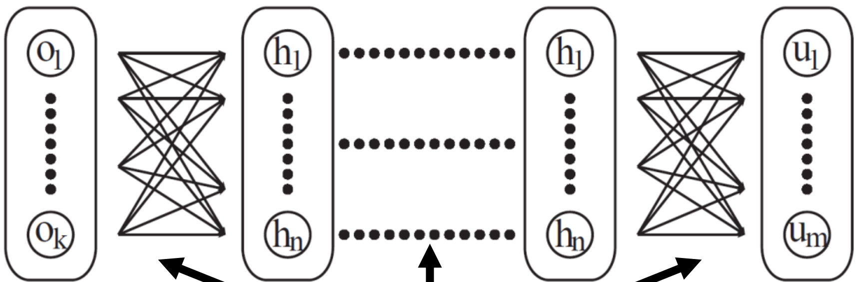


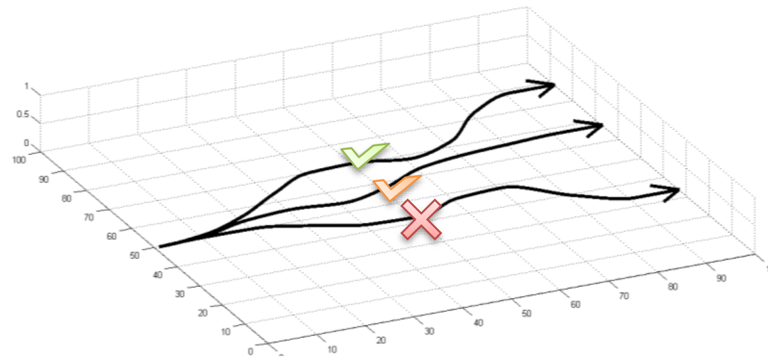
$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) - \text{control policy}$$

$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)

general-purpose neural network policy



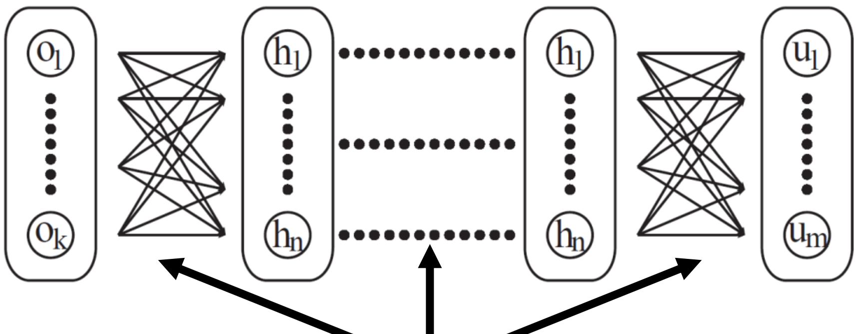


$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t\text{)}$$

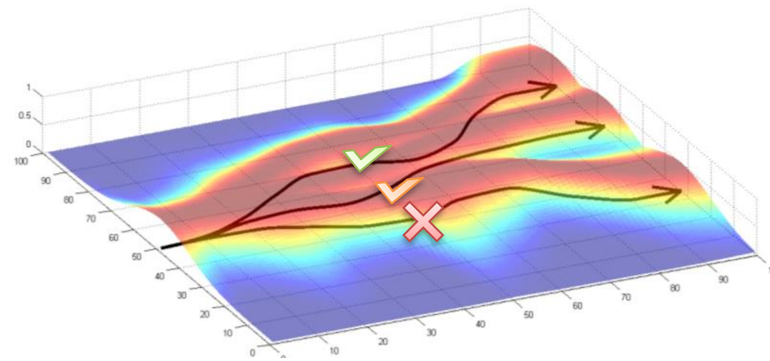
general-purpose neural network policy



θ

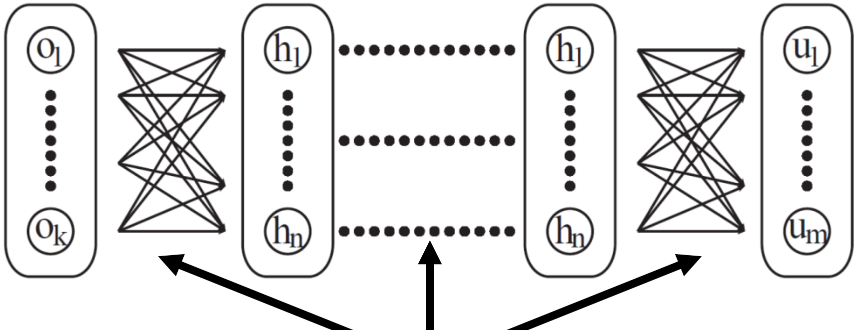
$$\theta = \arg\min_\theta E_{\pi_\theta}[\textstyle\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

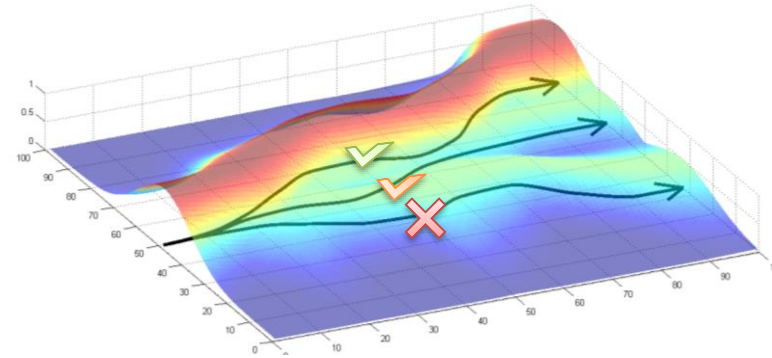
$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) - \text{control policy}$$



$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)

general-purpose neural network policy



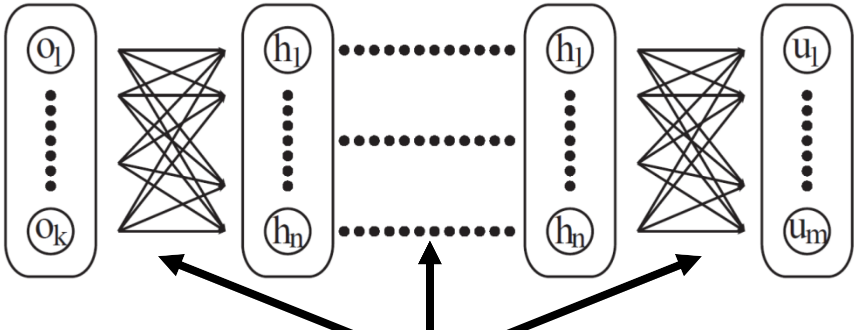


$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t)$$

general-purpose neural network policy

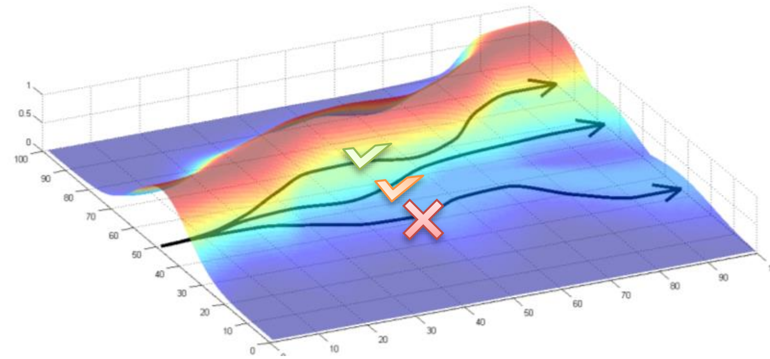


θ

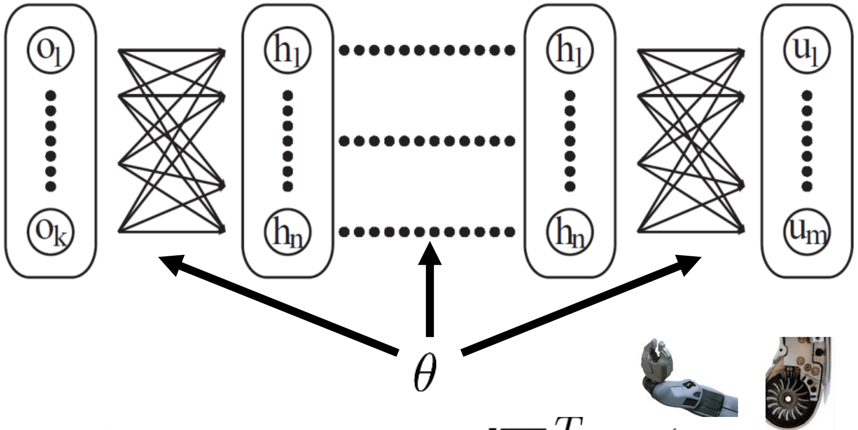
$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t)$$



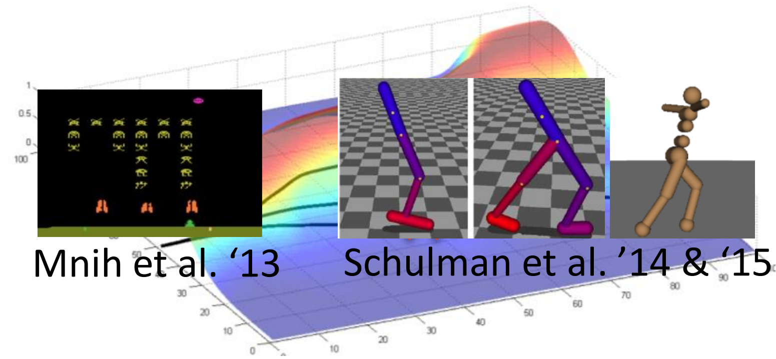
general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

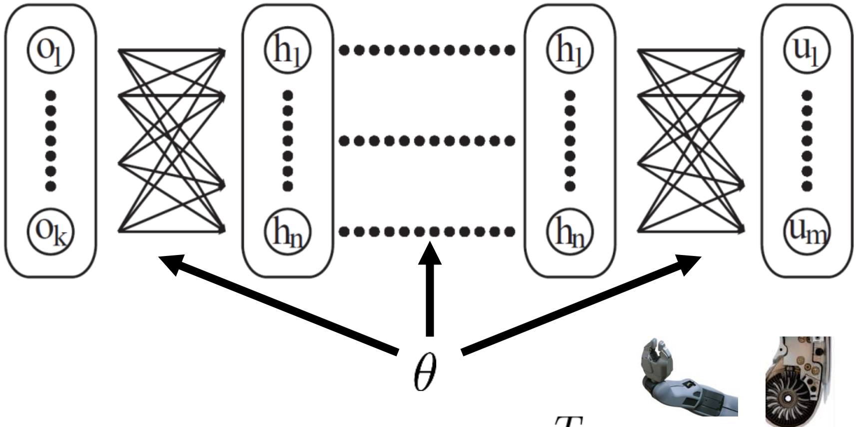
$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t)$$



Mnih et al. '13 Schulman et al. '14 & '15

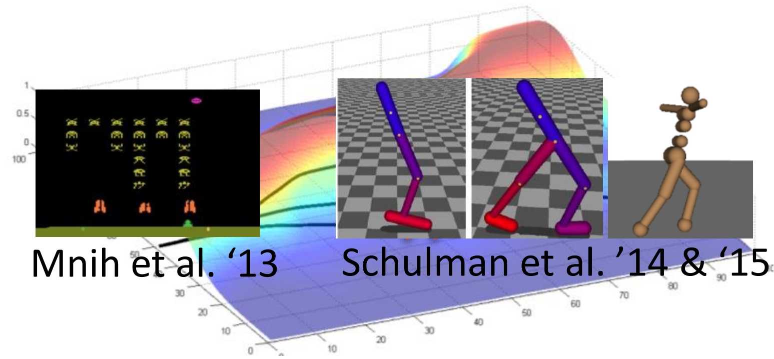
general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

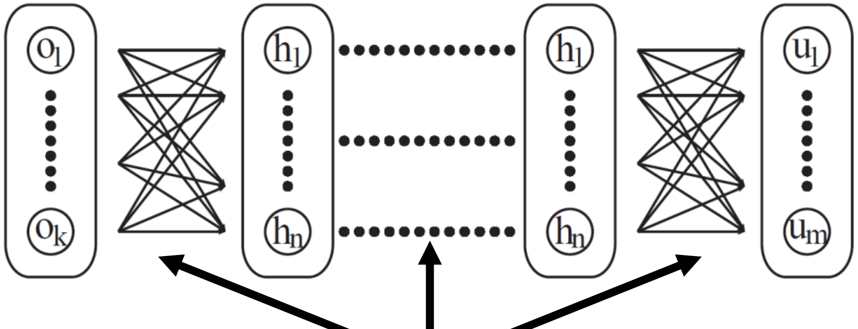
$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t\text{)}$$



Mnih et al. '13 Schulman et al. '14 & '15

policy search (RL)

general-purpose neural network policy

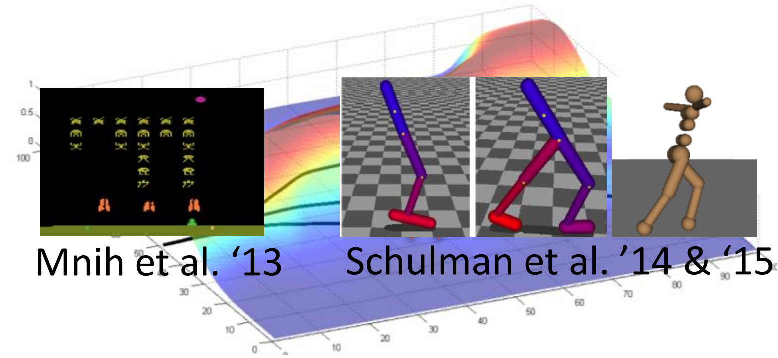


θ

$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t\text{)}$$

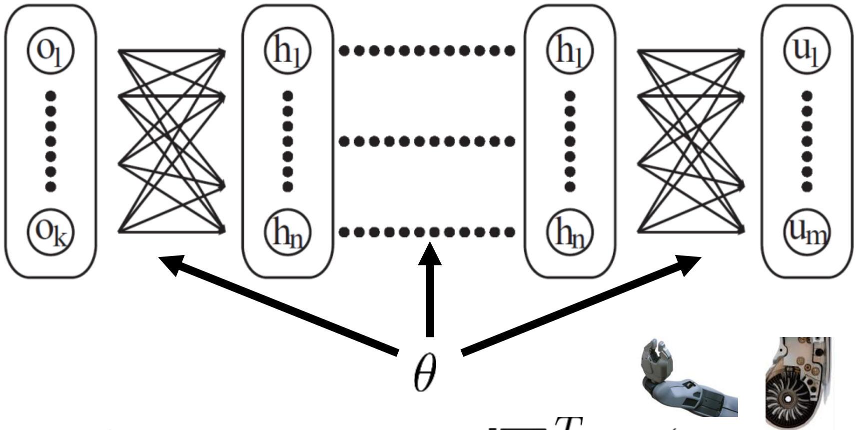


Mnih et al. '13 Schulman et al. '14 & '15

policy search (RL)

complex dynamics

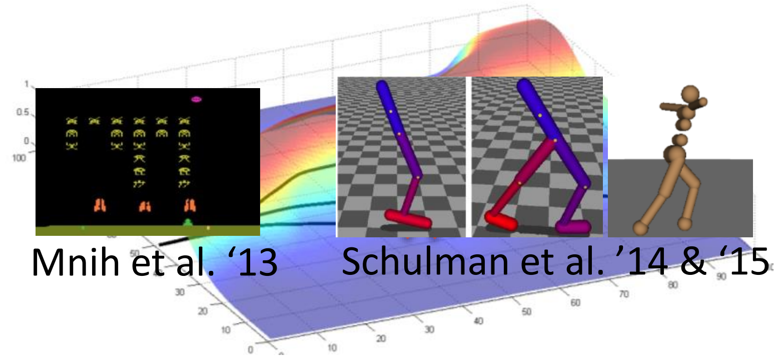
general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t\text{)}$$

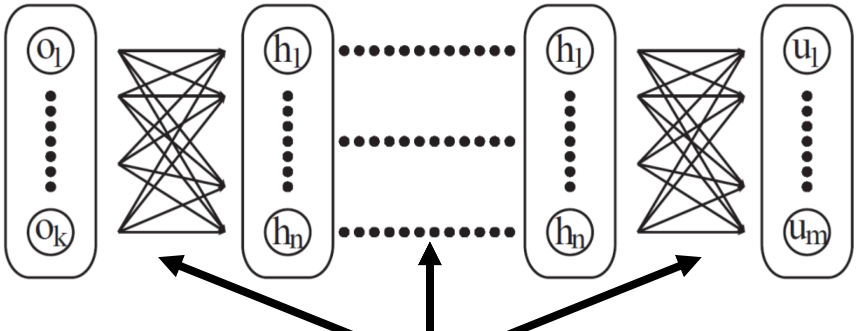


policy search (RL)

complex dynamics

complex policy

general-purpose neural network policy

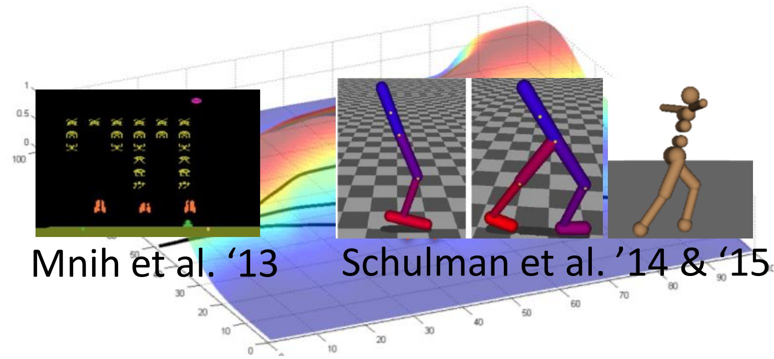


θ

$$\theta = \arg\min_\theta E_{\pi_\theta}[\textstyle\sum_{t=1}^T c(\mathbf{x}_t, \mathbf{u}_t)]$$

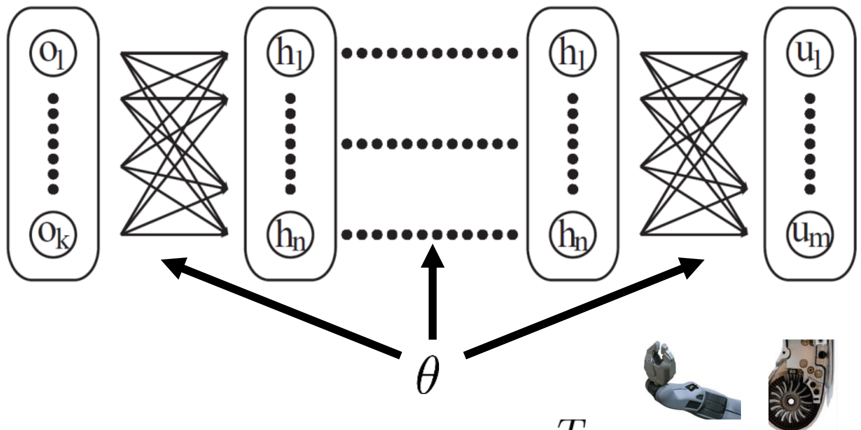
$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ – control policy

$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)



policy search (RL) complex dynamics complex policy HARD

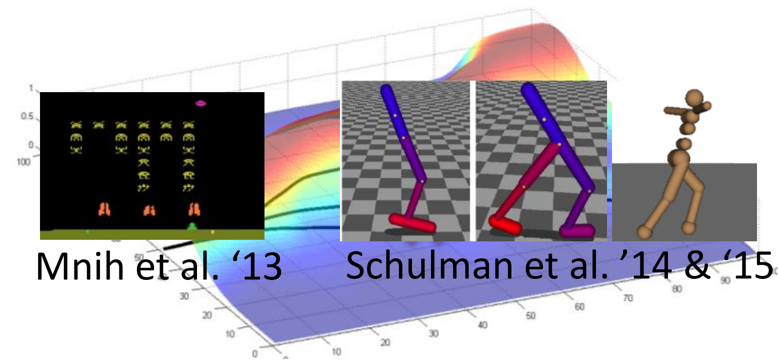
general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^T c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) - \text{control policy}$$

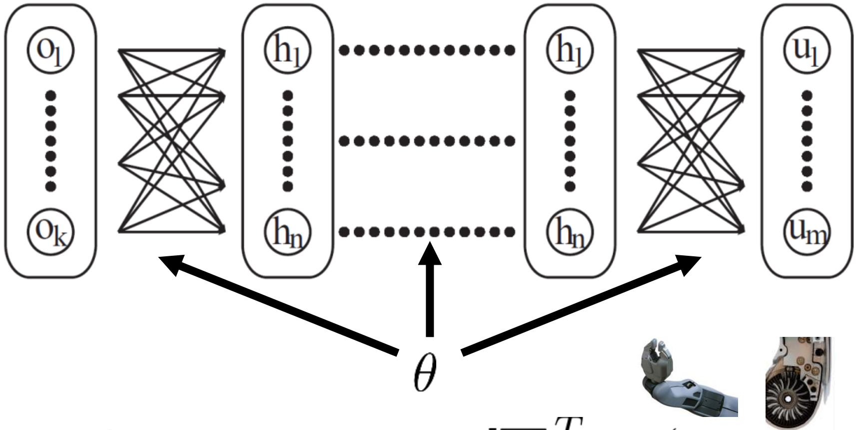
$$\mathbf{o}_t - \text{observation (may or may not be equal to } \mathbf{x}_t)$$



policy search (RL) complex dynamics complex policy HARD

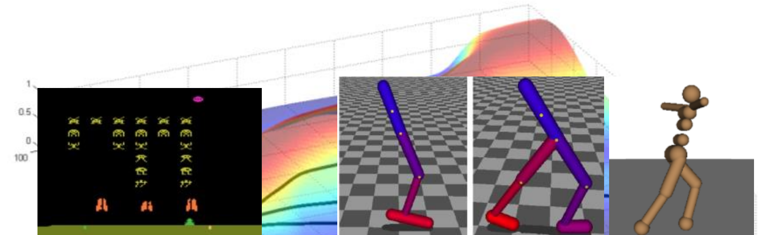
supervised learning

general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\textstyle\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) - \text{control policy}$$



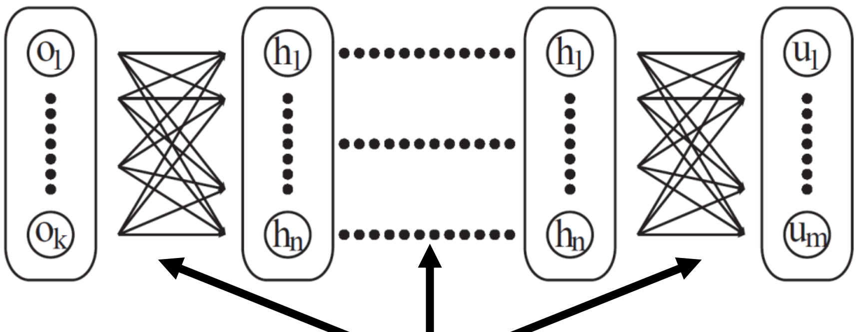
Mnih et al. '13 Schulman et al. '14 & '15

$$\mathbf{o}_t - \text{observation (may or may not be equal to } \mathbf{x}_t)$$

policy search (RL) complex dynamics complex policy HARD

supervised learning ~~complex dynamics~~

general-purpose neural network policy

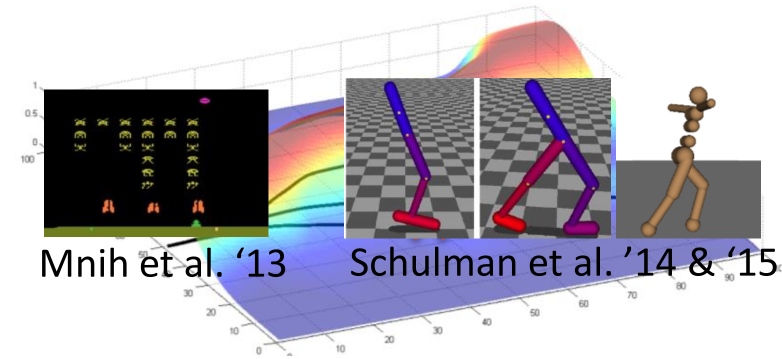


θ

$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) - \text{control policy}$$

$$\mathbf{o}_t - \text{observation (may or may not be equal to } \mathbf{x}_t)$$

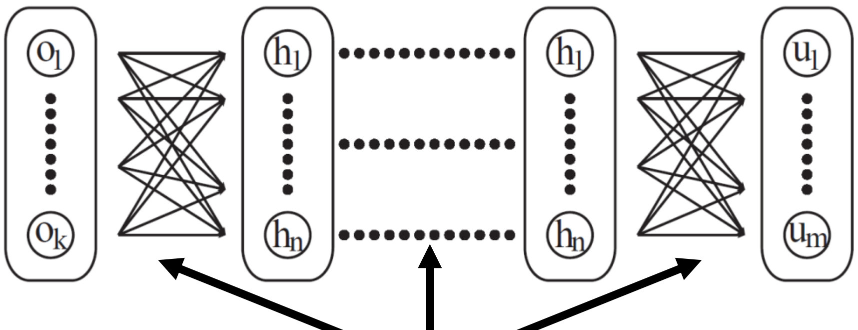


Mnih et al. '13 Schulman et al. '14 & '15

policy search (RL) complex dynamics complex policy HARD

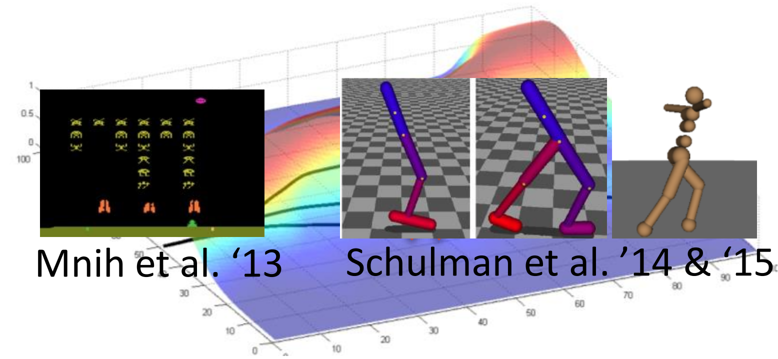
supervised learning ~~complex dynamics~~ complex policy

general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

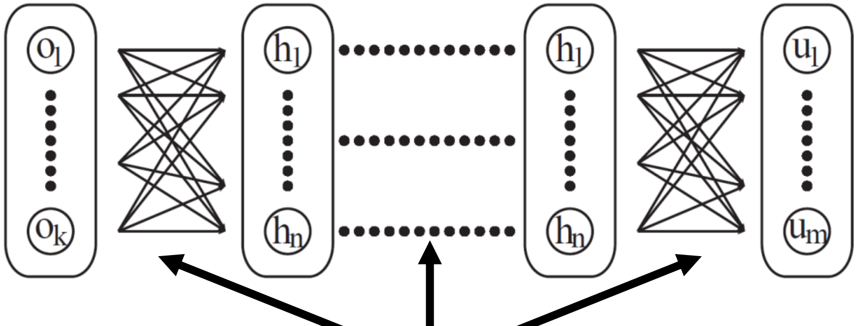
$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$



$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t)$$

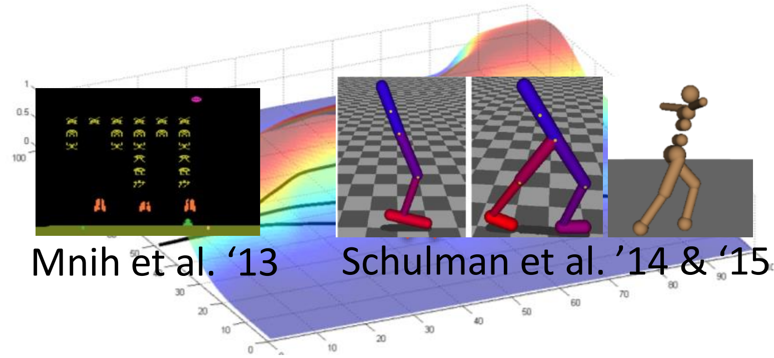
policy search (RL)	complex dynamics	complex policy	HARD
supervised learning	~~complex dynamics~~	complex policy	EASY

general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}[\textstyle\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

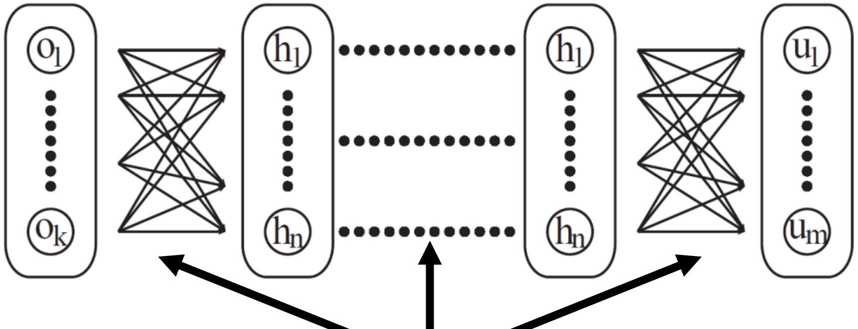


Mnih et al. '13 Schulman et al. '14 & '15

$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t)$$

policy search (RL)	complex dynamics	complex policy	HARD
supervised learning	~~complex dynamics~~	complex policy	EASY
optimal control			

general-purpose neural network policy

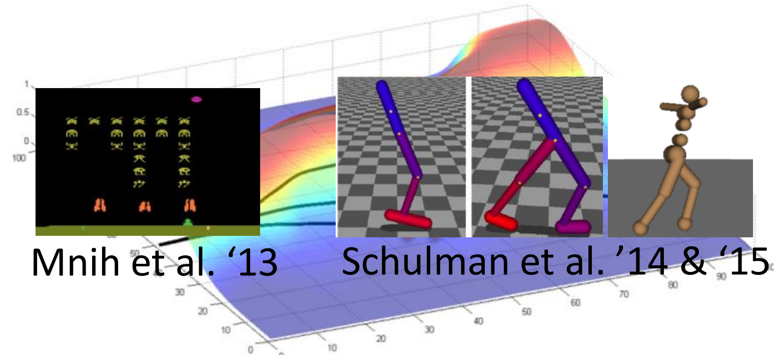


θ

$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ – control policy}$$

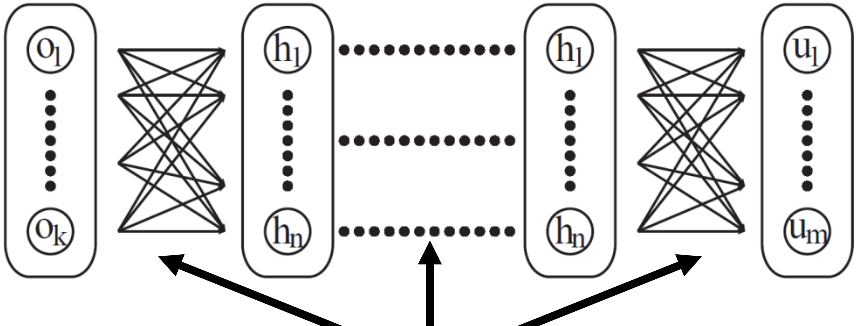
$$\mathbf{o}_t \text{ – observation (may or may not be equal to } \mathbf{x}_t)$$

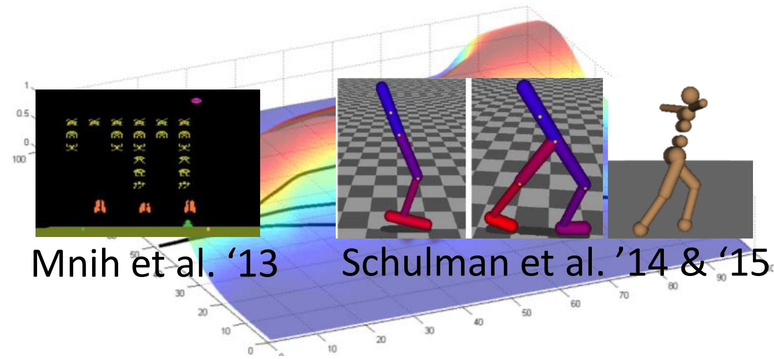


Mnih et al. '13 Schulman et al. '14 & '15

policy search (RL)	complex dynamics	complex policy	HARD
supervised learning	~~complex dynamics~~	complex policy	EASY
optimal control	complex dynamics		

general-purpose neural network policy





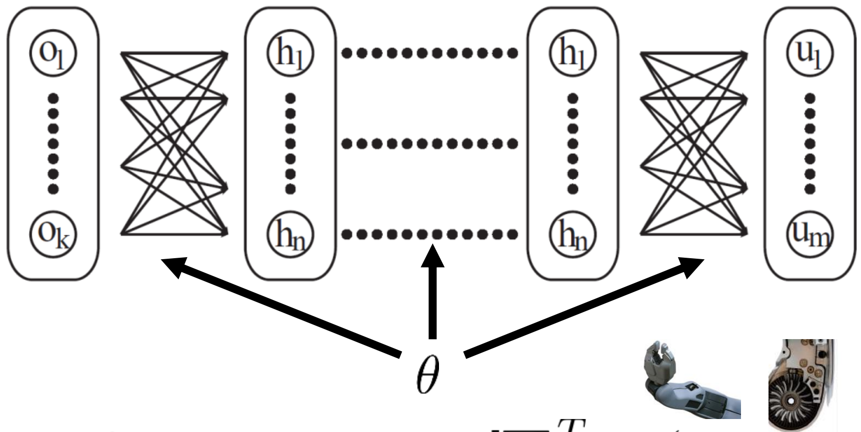
$$\theta = \arg\min_\theta E_{\pi_\theta}[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)]$$

$$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t) \text{ -- control policy}$$

$$\mathbf{o}_t \text{ -- observation (may or may not be equal to } \mathbf{x}_t\text{)}$$

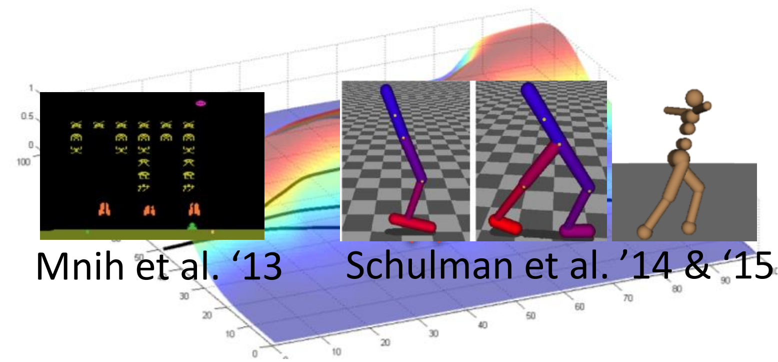
policy search (RL)	complex dynamics	complex policy	HARD
supervised learning	~~complex dynamics~~	complex policy	EASY
optimal control	complex dynamics	~~complex policy~~	

general-purpose neural network policy



$$\theta = \arg\min_\theta E_{\pi_\theta}\left[\sum_{t=1}^{T} c(\mathbf{x}_t, \mathbf{u}_t)\right]$$

$\pi_\theta(\mathbf{u}_t|\mathbf{o}_t)$ – control policy

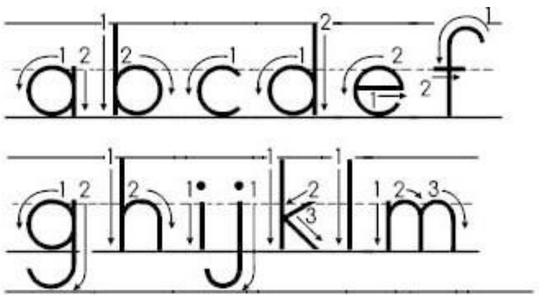


$\mathbf{o}_t$ – observation (may or may not be equal to $\mathbf{x}_t$)

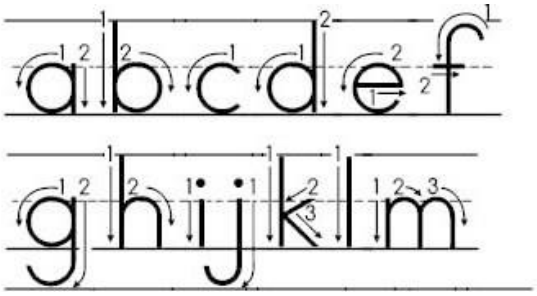
policy search (RL)	complex dynamics	complex policy	HARD
supervised learning	~~complex dynamics~~	complex policy	EASY
optimal control	complex dynamics	~~complex policy~~	EASY

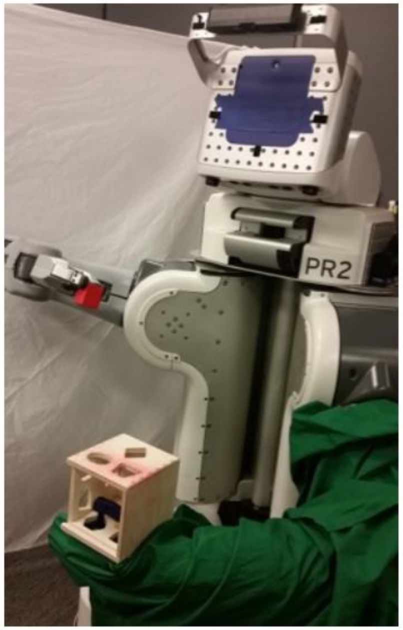
1. break up the task:
   separately solve N
   different task instances

1. break up the task: separately solve N different task instances

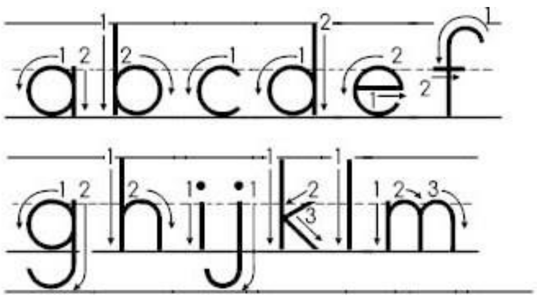


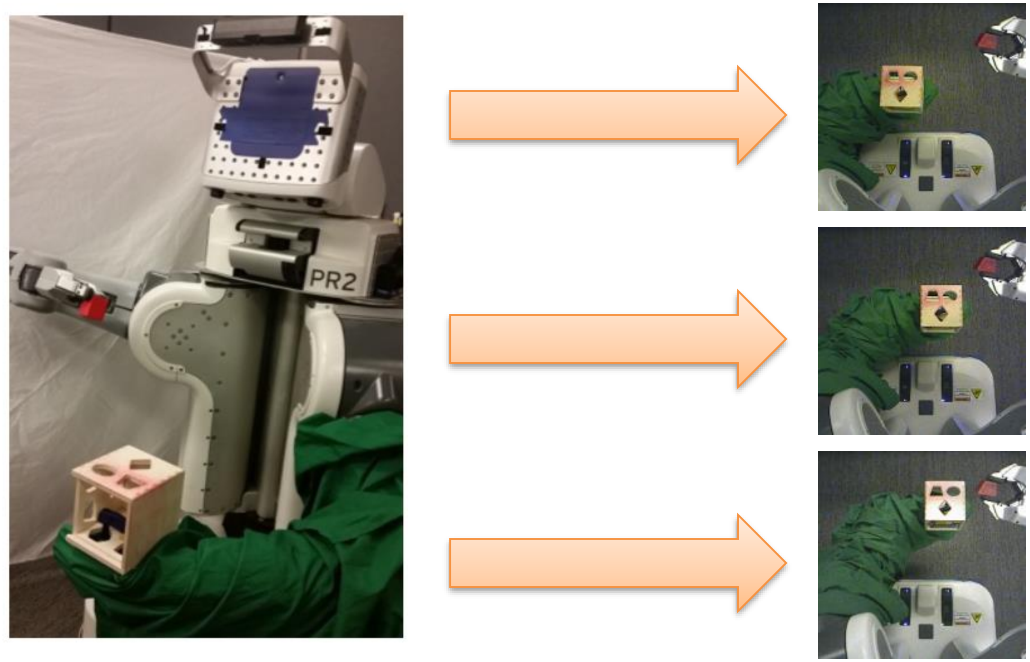
1. break up the task: separately solve N different task instances



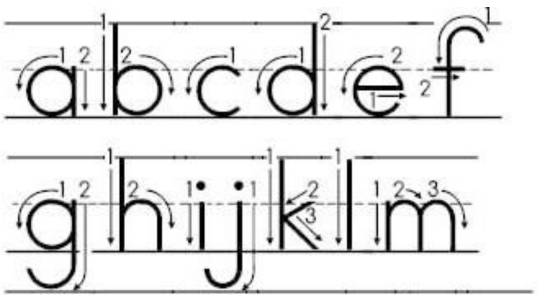


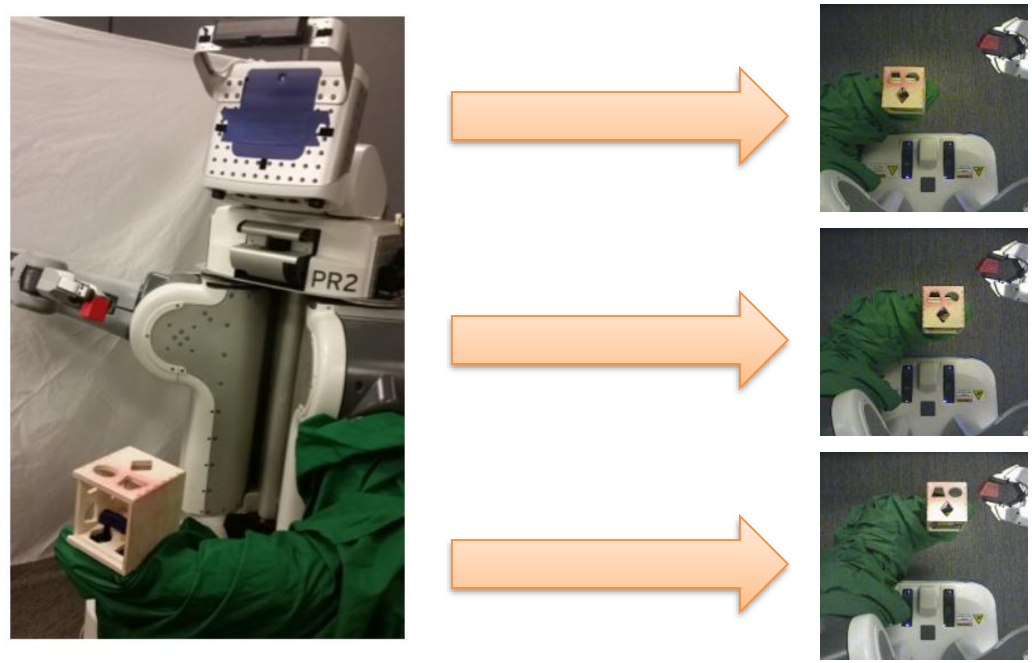
1. break up the task: separately solve N different task instances





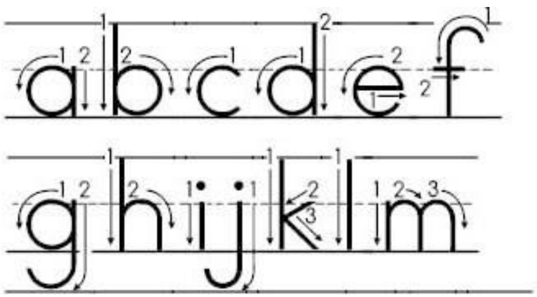
1. break up the task: separately solve N different task instances

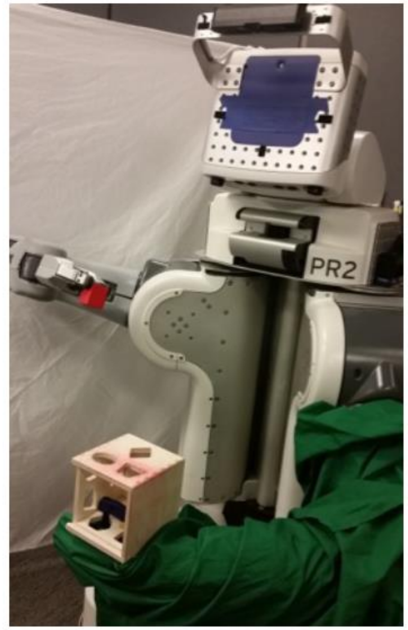


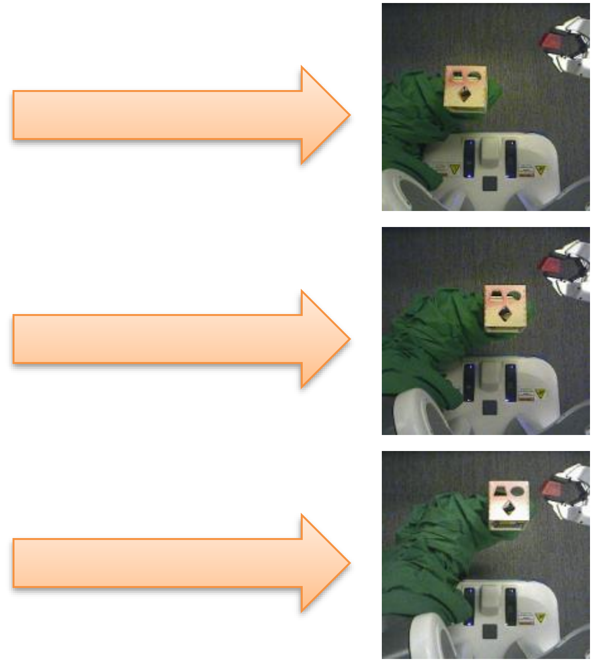


2. use supervised learning

1. break up the task: separately solve N different task instances

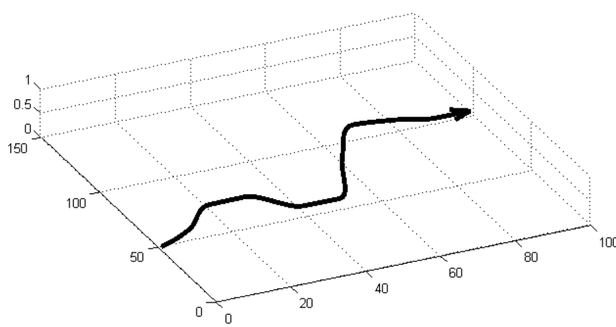




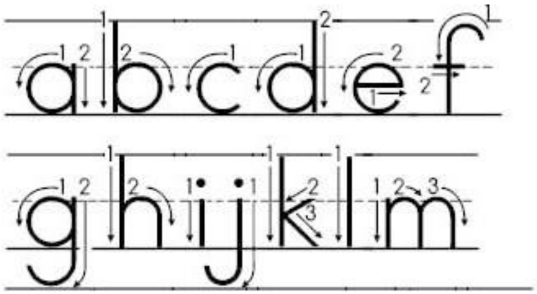


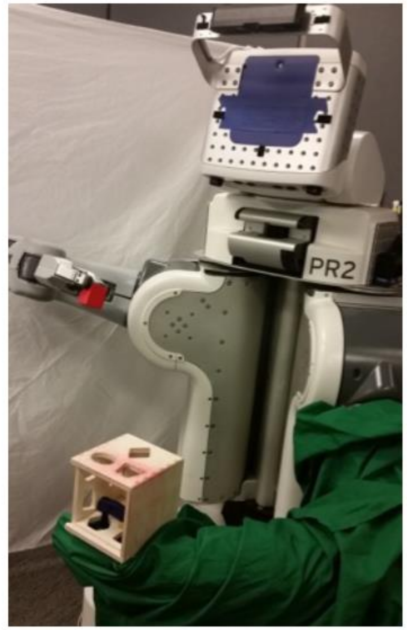
2. use supervised learning

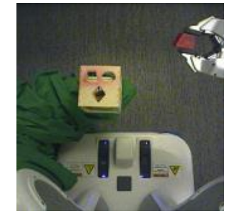
trajectory-centric RL
(fully observed)

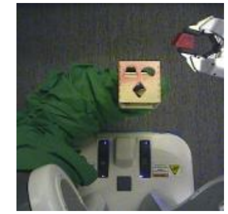


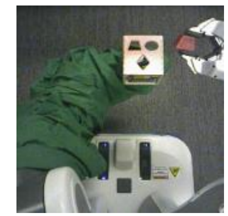
1. break up the task: separately solve N different task instances





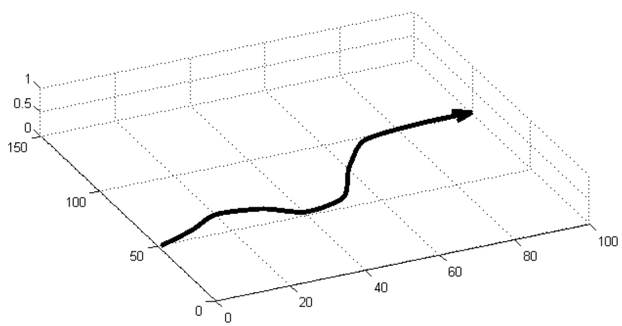




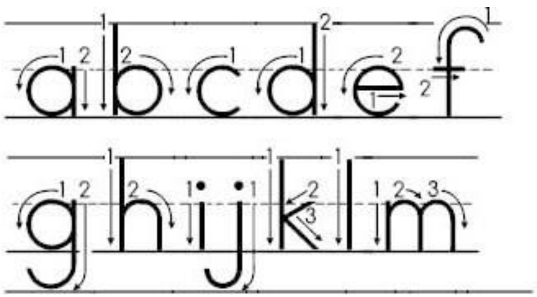


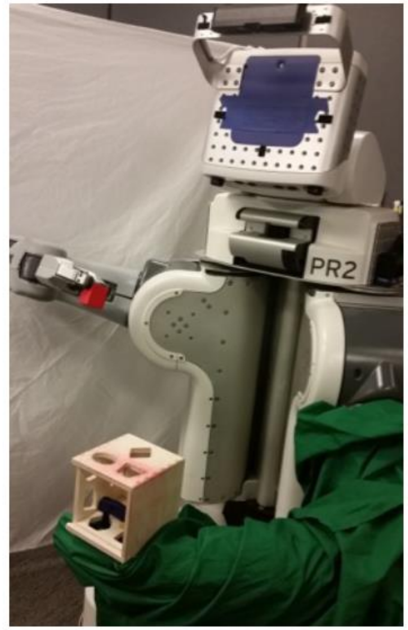
2. use supervised learning

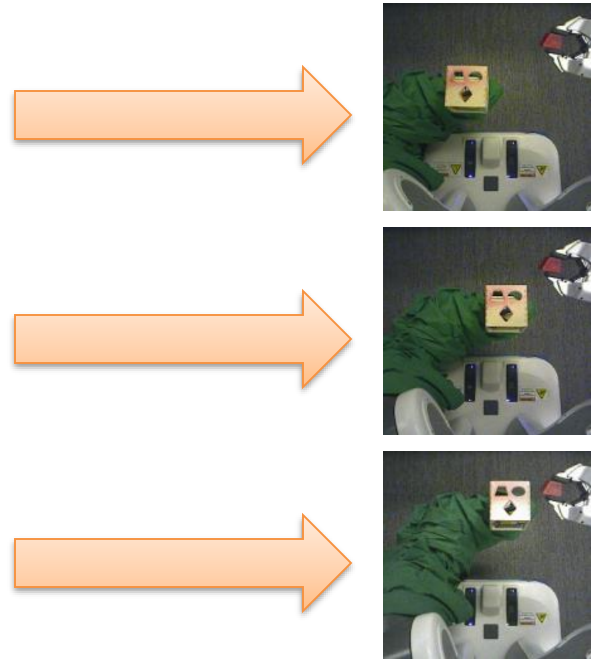
trajectory-centric RL
(fully observed)



1. break up the task: separately solve N different task instances

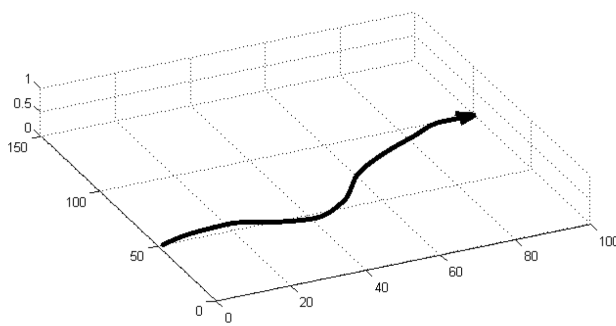




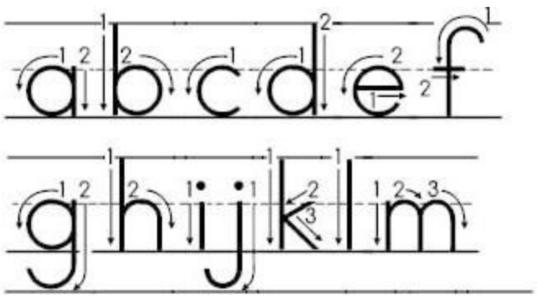


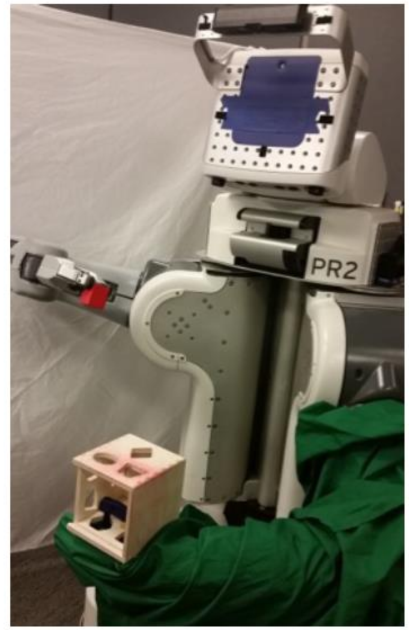
2. use supervised learning

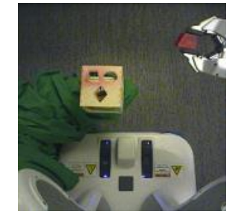
trajectory-centric RL
(fully observed)

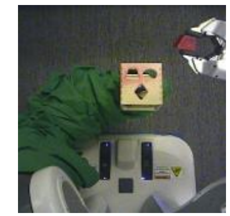


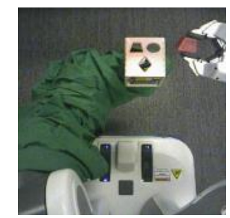
1. break up the task: separately solve N different task instances





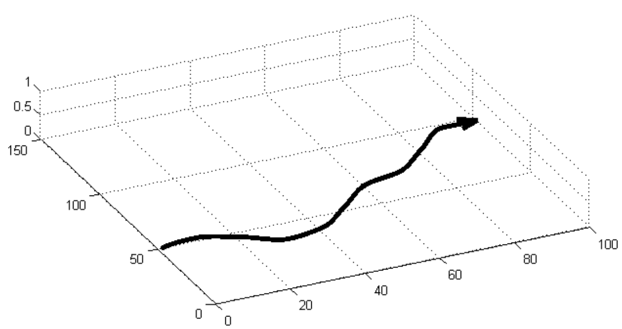




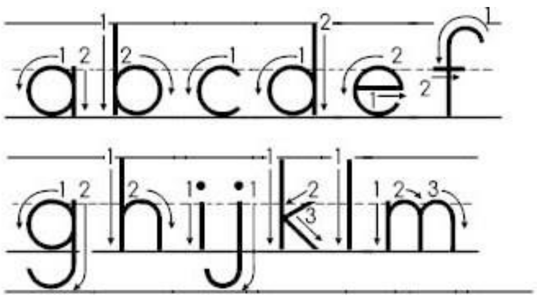


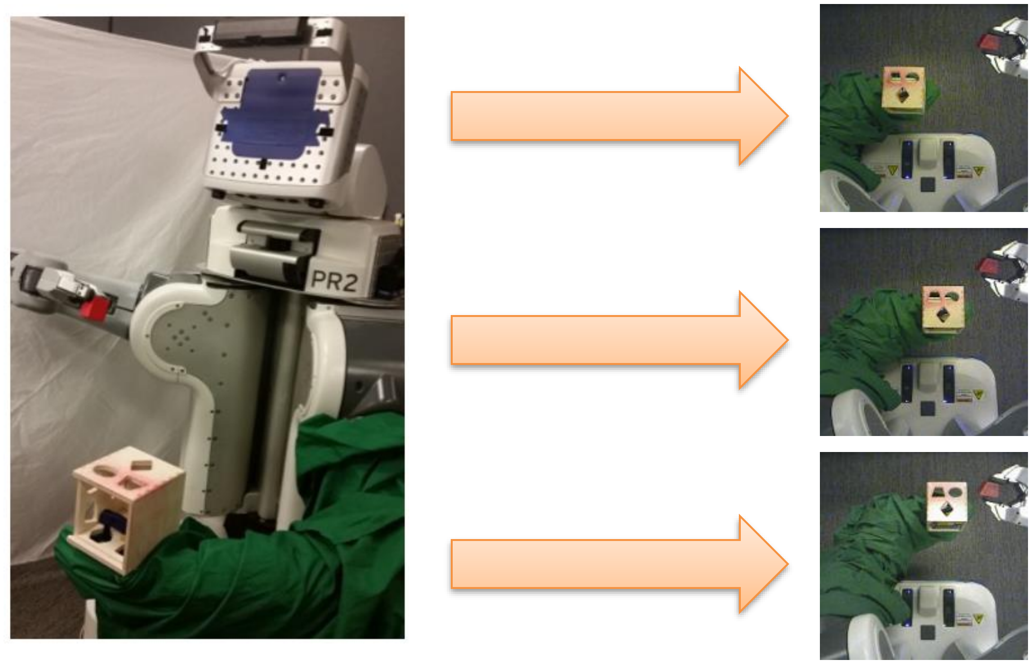
2. use supervised learning

trajectory-centric RL
(fully observed)



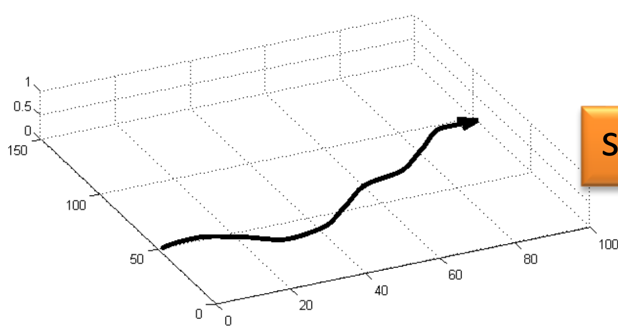
1. break up the task: separately solve N different task instances

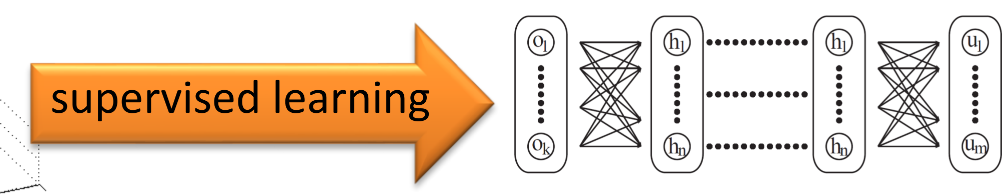




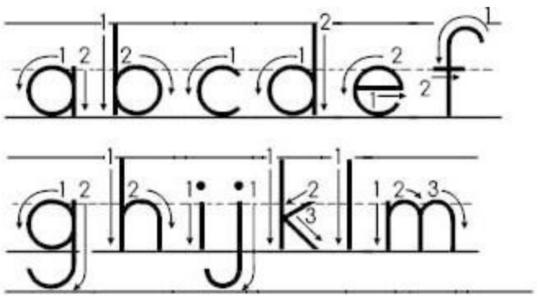
2. use supervised learning

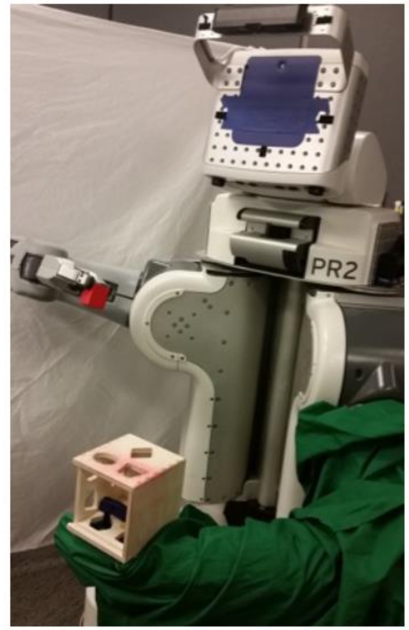
trajectory-centric RL
(fully observed)

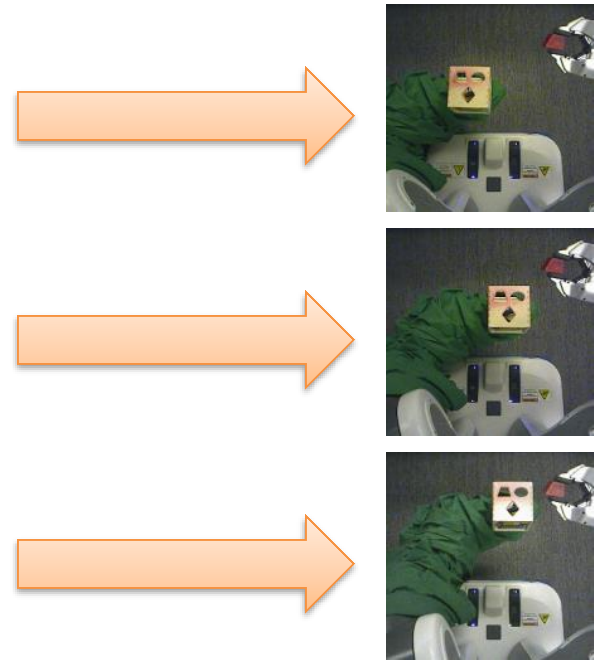




1. break up the task: separately solve N different task instances

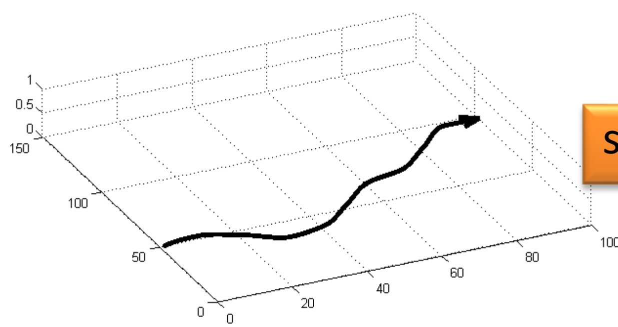




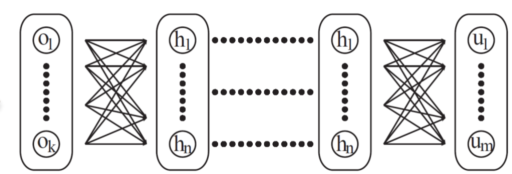


2. use supervised learning

trajectory-centric RL
(fully observed)



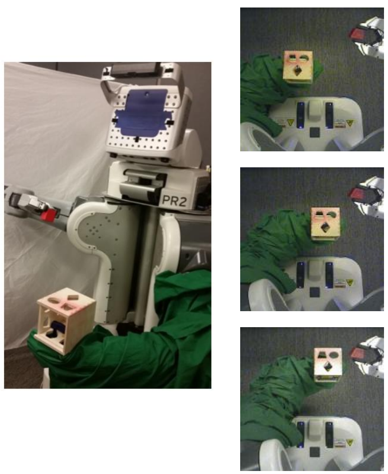
state to **action**

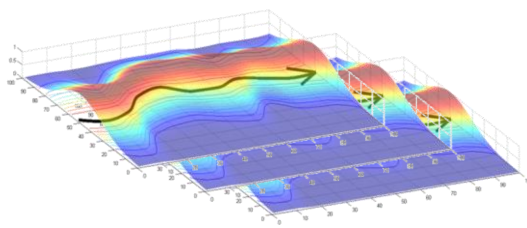


observation to **action**

Guided Policy Search

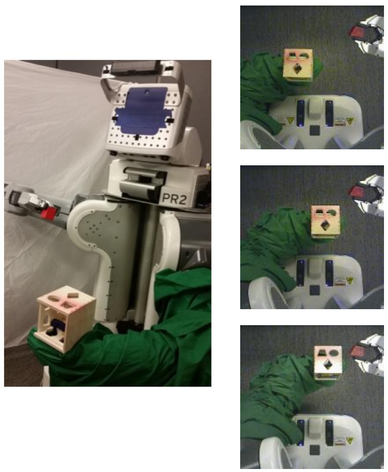
Guided Policy Search

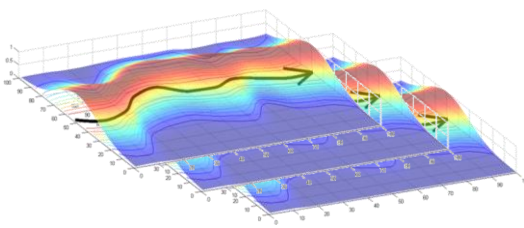




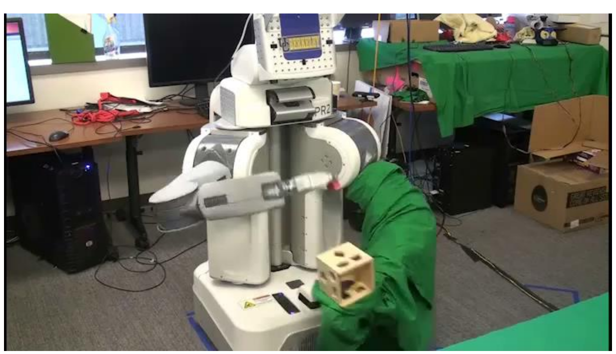
trajectory-centric RL

Guided Policy Search

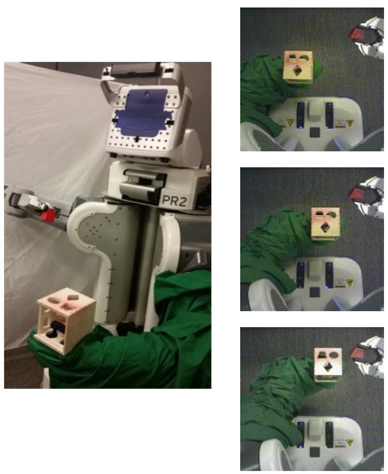


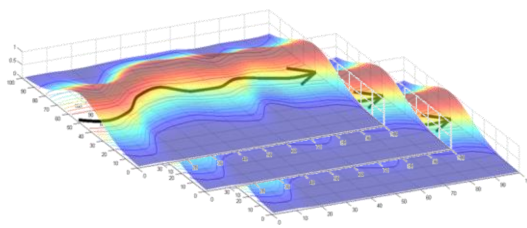


trajectory-centric RL

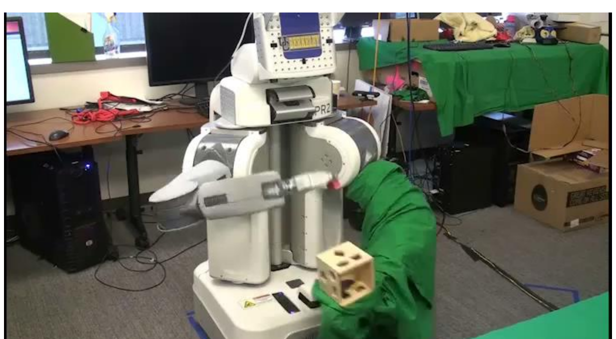


Guided Policy Search

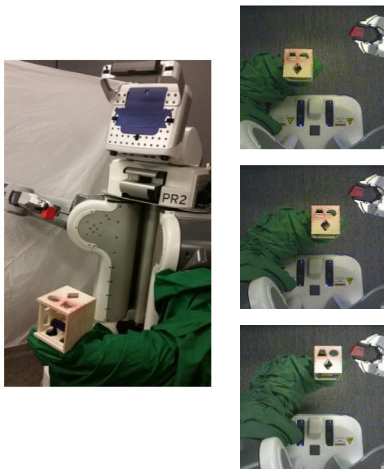


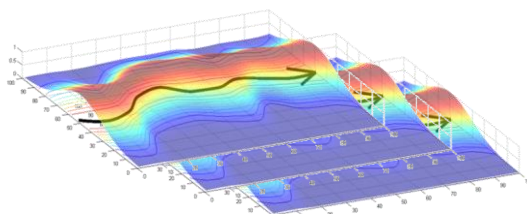


trajectory-centric RL

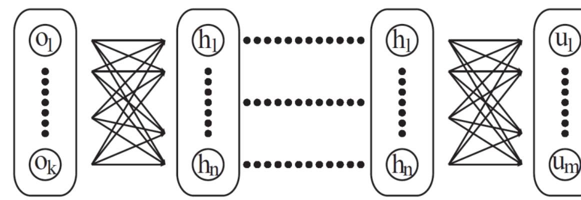


Guided Policy Search

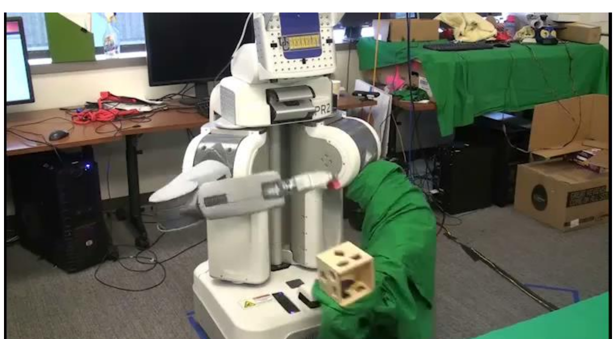




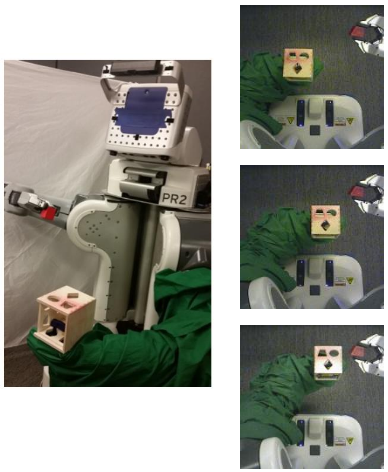
trajectory-centric RL

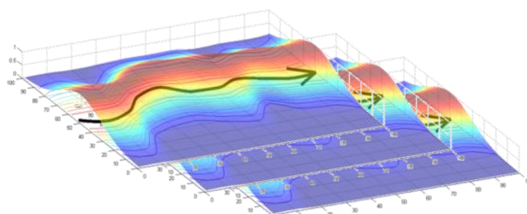


supervised learning

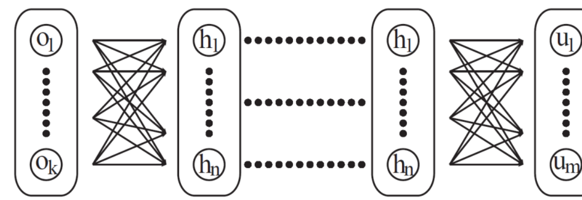


Guided Policy Search

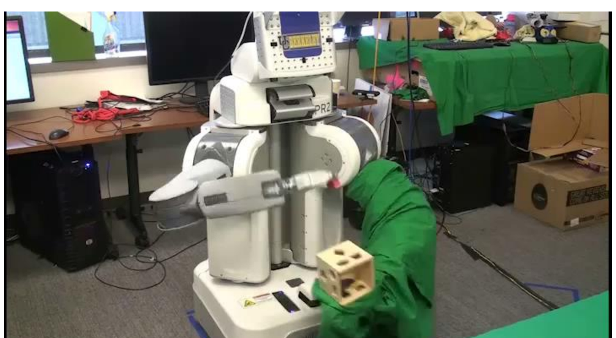




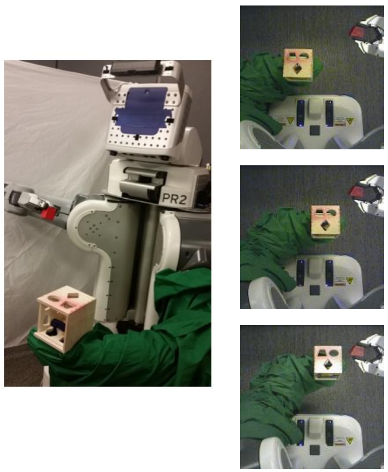
trajectory-centric RL

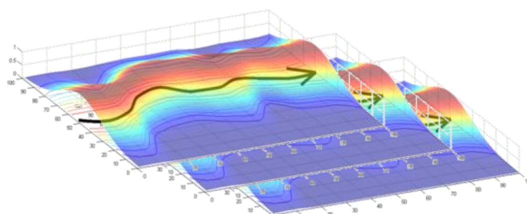


supervised learning

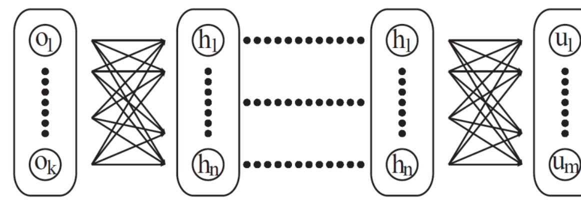


Guided Policy Search

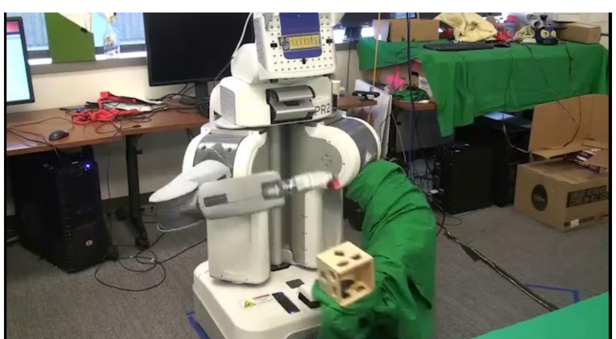




trajectory-centric RL



supervised learning



$$\min_{\theta} E_{\pi_\theta}[c(\tau)]$$

$$\min_{\theta} \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

$$\min_{\theta} \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

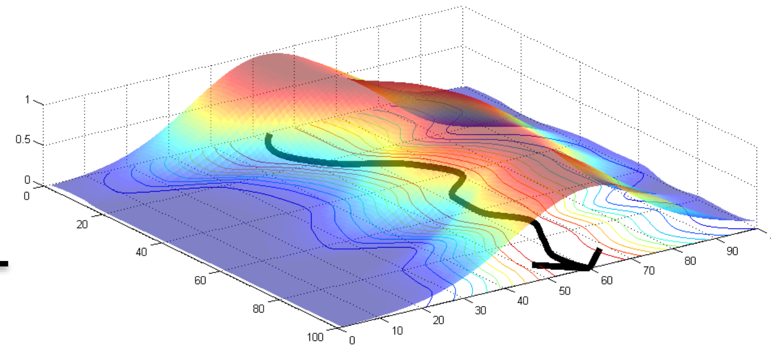
$$\min_{\theta, p(\tau)} \; E_p[c(\tau)]$$

$$s.t. \;\; \pi_\theta(\mathbf{u}_t|\mathbf{o}(\mathbf{x}_t)) = p(\mathbf{u}_t|\mathbf{x}_t) \;\; \forall t, \mathbf{x}_t, \mathbf{u}_t$$

$$\min_{\theta} \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

$$\min_{\theta, p(\tau)} \; E_p[c(\tau)]$$

trajectory distribution(s)

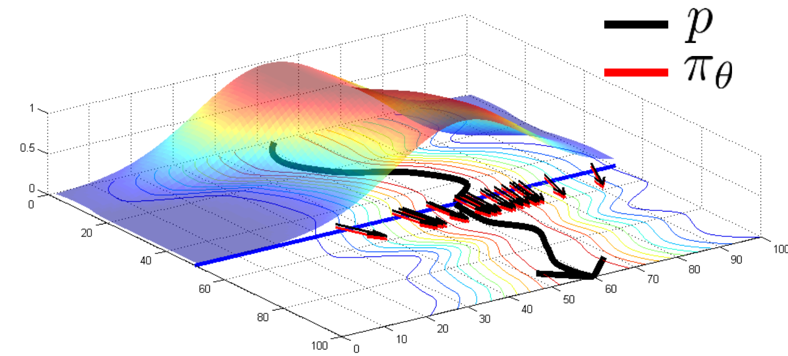


$$s.t. \;\; \pi_\theta(\mathbf{u}_t|\mathbf{o}(\mathbf{x}_t)) = p(\mathbf{u}_t|\mathbf{x}_t) \;\; \forall t, \mathbf{x}_t, \mathbf{u}_t$$

$$\min_{\theta} \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

$$\min_{\theta, p(\tau)} \; E_p[c(\tau)]$$

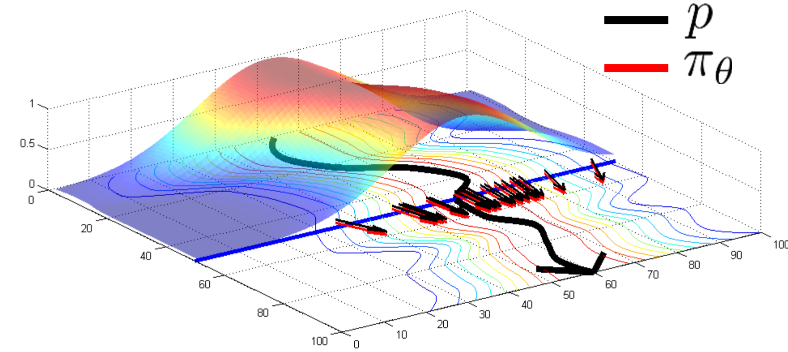
$$s.t. \;\; \pi_\theta(\mathbf{u}_t | \mathbf{o}(\mathbf{x}_t)) = p(\mathbf{u}_t | \mathbf{x}_t) \;\; \forall t, \mathbf{x}_t, \mathbf{u}_t$$



$$\min_\theta \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

$$\min_{\theta, p(\tau)} \ E_p[c(\tau)]$$

$$s.t. \ \ \pi_\theta(\mathbf{u}_t|\mathbf{o}(\mathbf{x}_t)) = p(\mathbf{u}_t|\mathbf{x}_t) \ \ \forall t, \mathbf{x}_t, \mathbf{u}_t$$

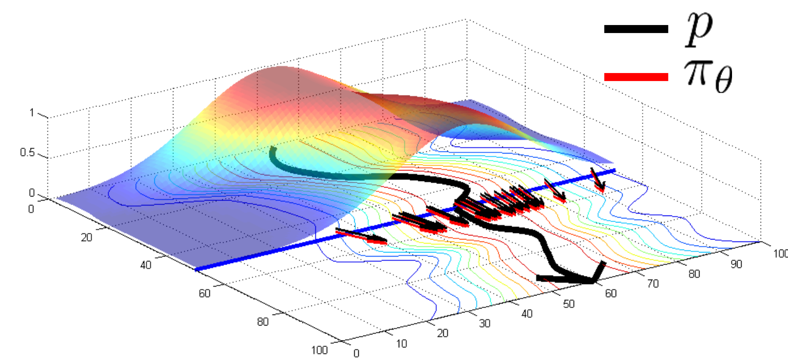


solve using Bregman ADMM (BADMM), a type of dual decomposition method

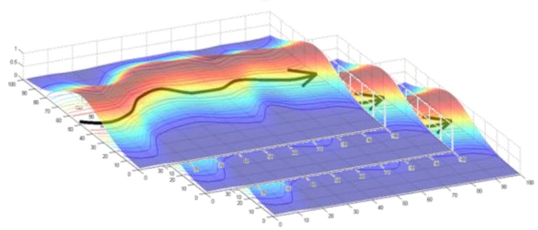
$$\min_{\theta} \overbrace{E_{\pi_\theta}}^{\text{expectation under current policy}}[c(\tau)]$$

$$\min_{\theta, p(\tau)} \; E_p[c(\tau)]$$

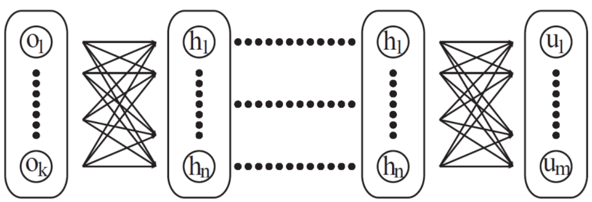
$$s.t. \;\; \pi_\theta(\mathbf{u}_t|\mathbf{o}(\mathbf{x}_t)) = p(\mathbf{u}_t|\mathbf{x}_t) \;\; \forall t, \mathbf{x}_t, \mathbf{u}_t$$



solve using Bregman ADMM (BADMM), a type of dual decomposition method



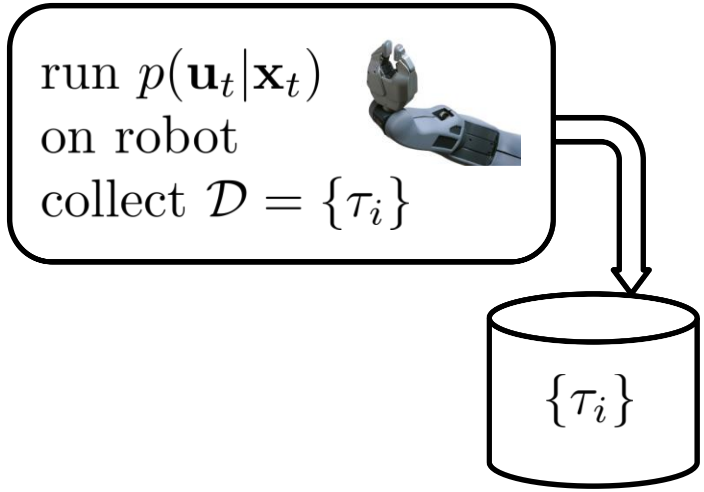
trajectory-centric RL



supervised learning

run $p(\mathbf{u}_t|\mathbf{x}_t)$
on robot
collect $\mathcal{D} = \{\tau_i\}$

[see L. et al. NIPS ‘14 for details]

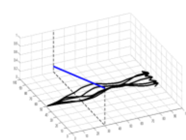


[see L. et al. NIPS ‘14 for details]

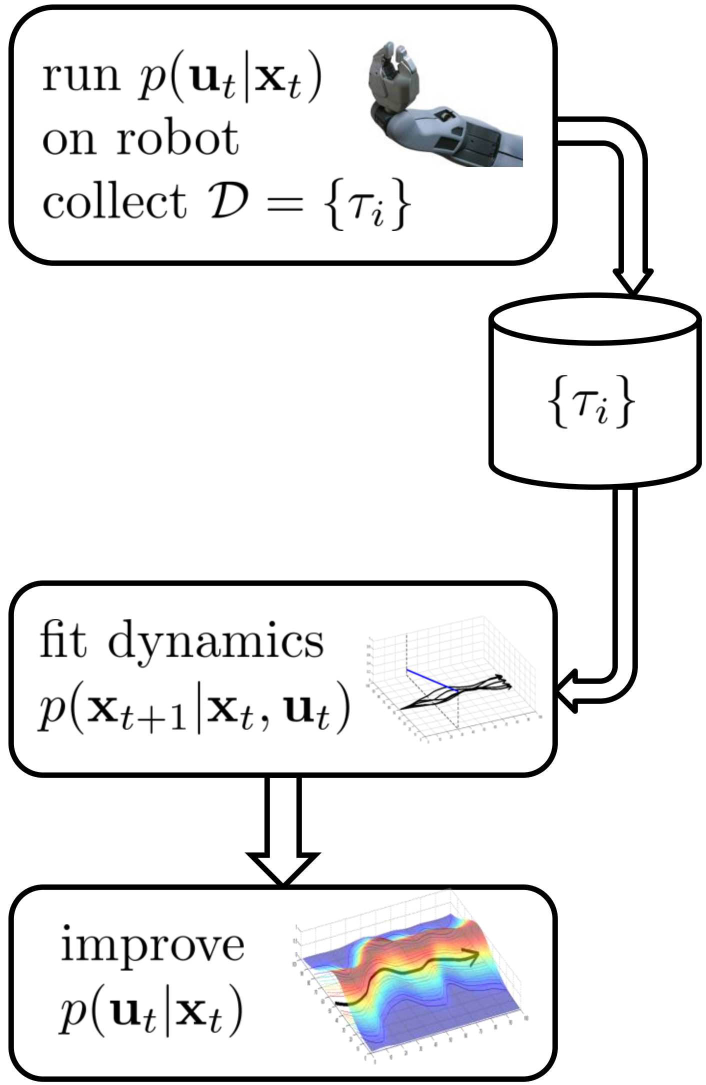
run $p(\mathbf{u}_t|\mathbf{x}_t)$
on robot
collect $\mathcal{D} = \{\tau_i\}$

$\{\tau_i\}$

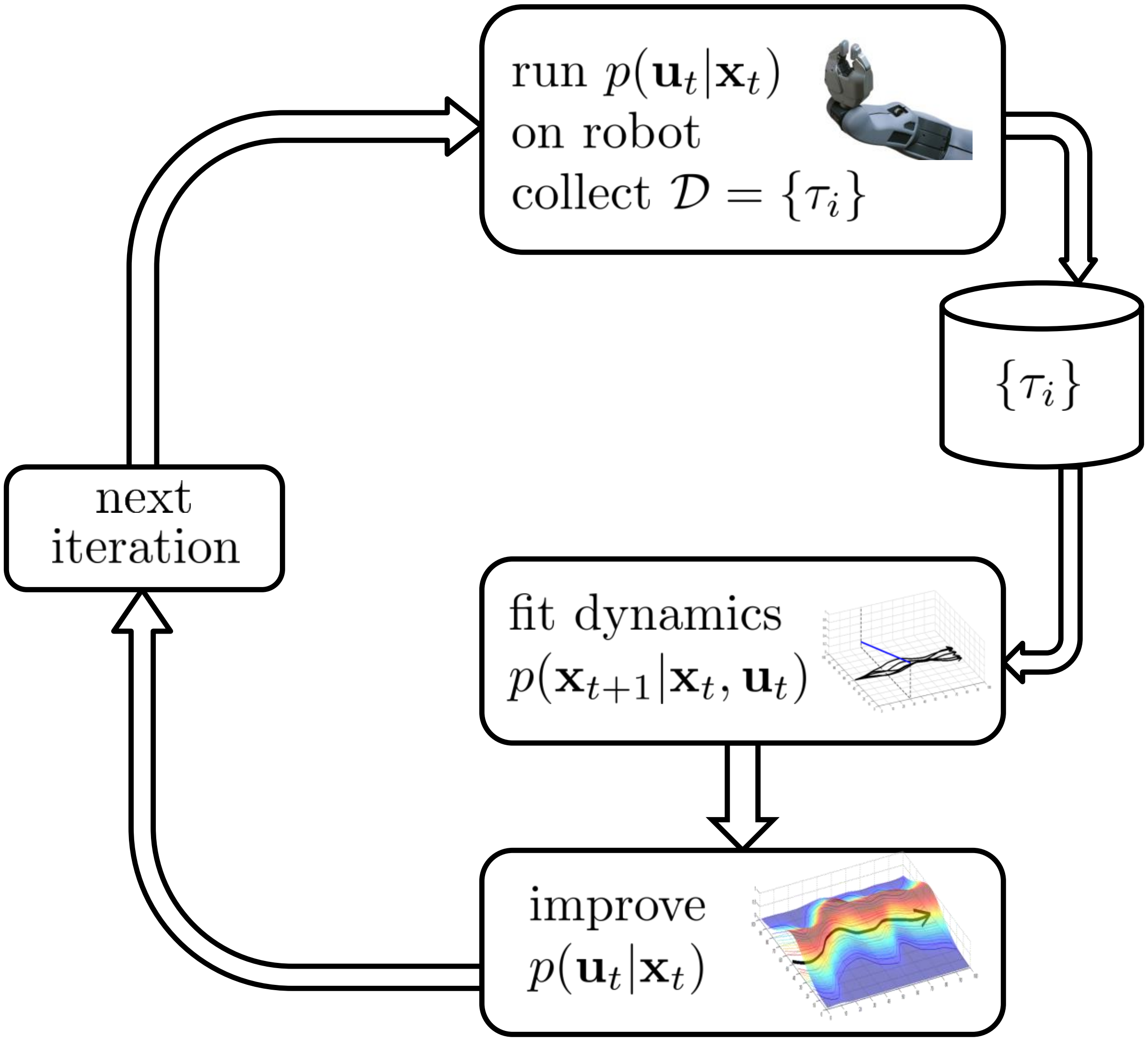
fit dynamics
$p(\mathbf{x}_{t+1}|\mathbf{x}_t, \mathbf{u}_t)$



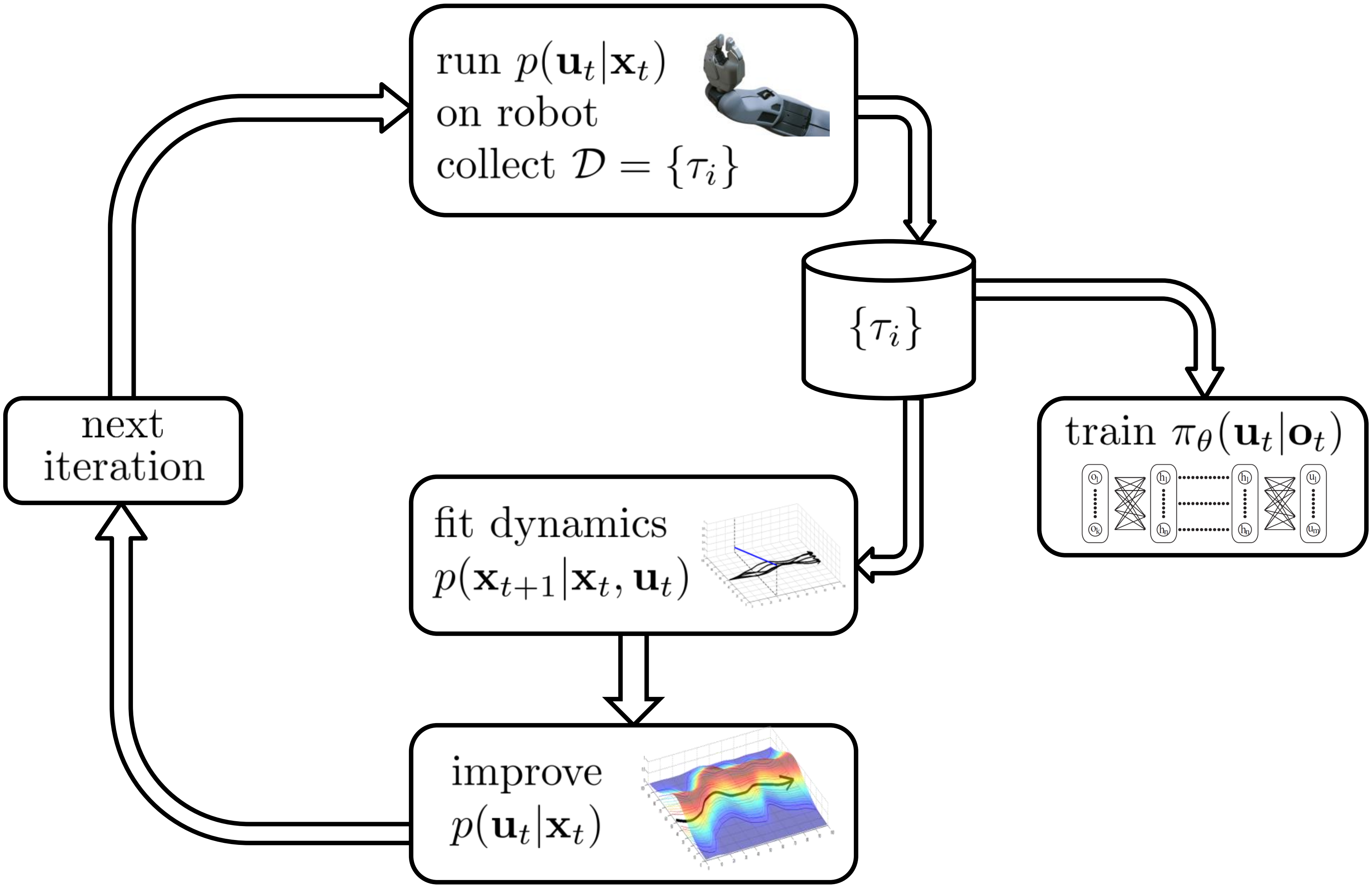
[see L. et al. NIPS '14 for details]



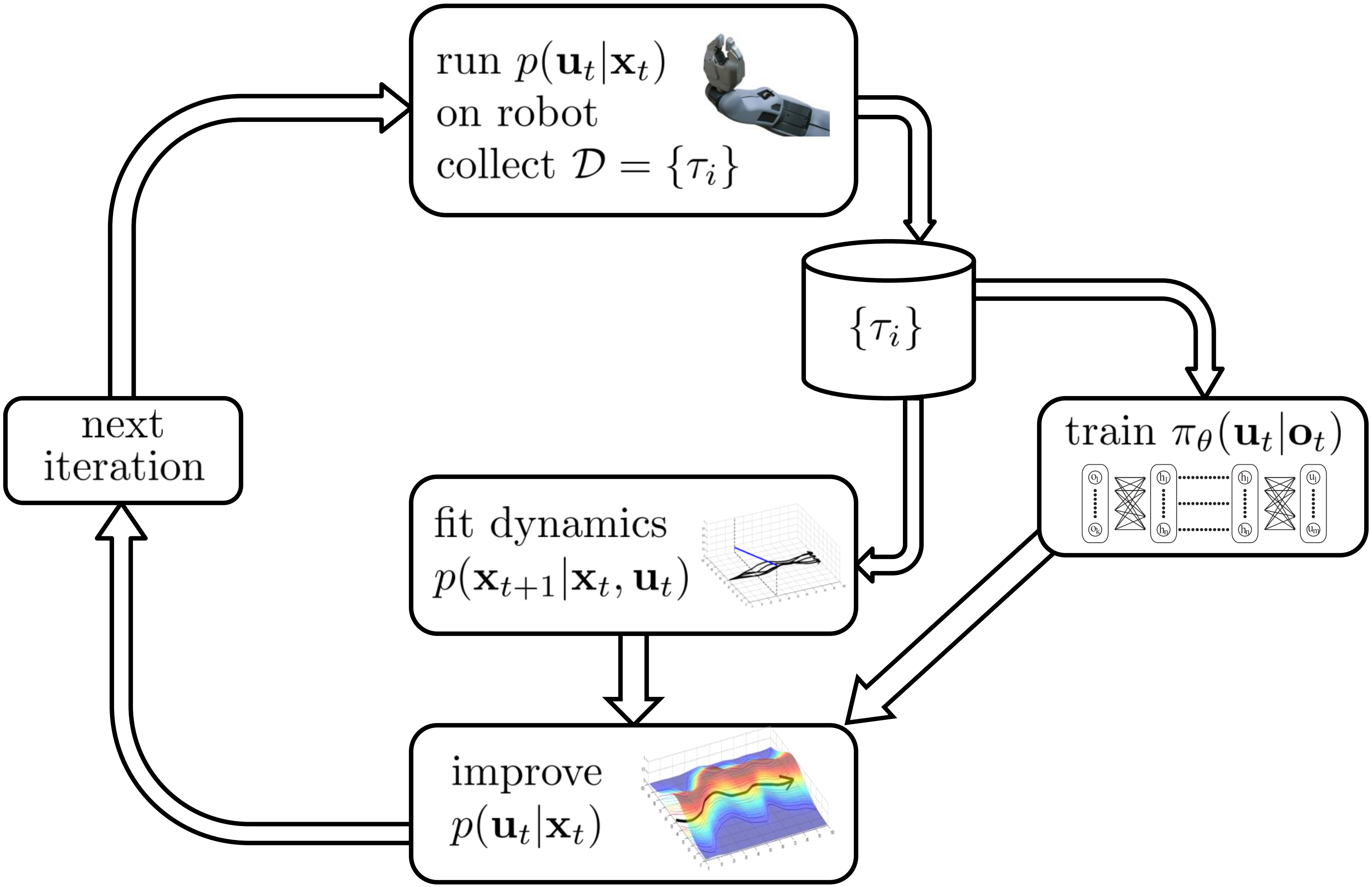
[see L. et al. NIPS '14 for details]



[see L. et al. NIPS '14 for details]



[see L. et al. NIPS ‘14 for details]

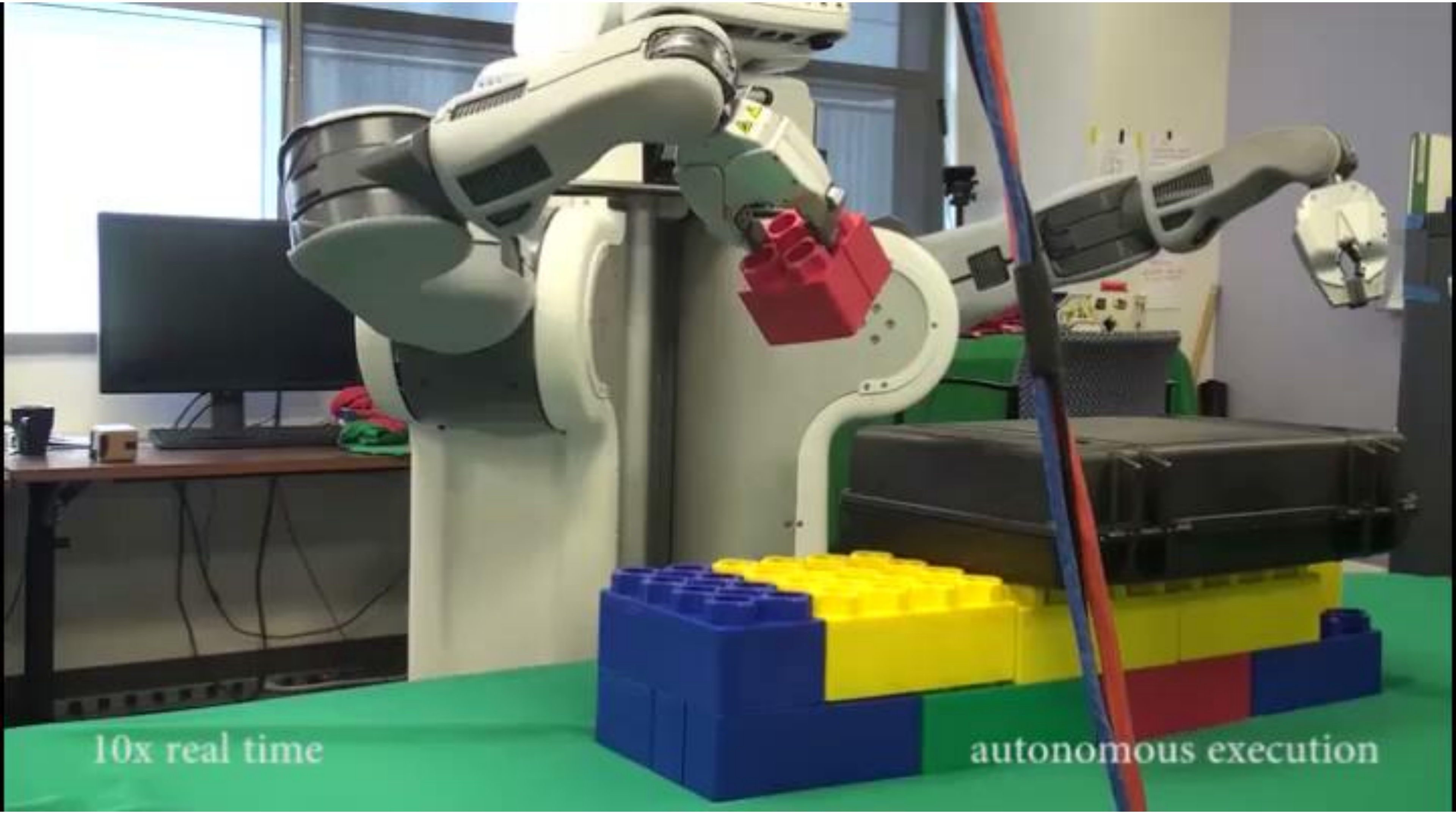


[see L. et al. NIPS '14 for details]

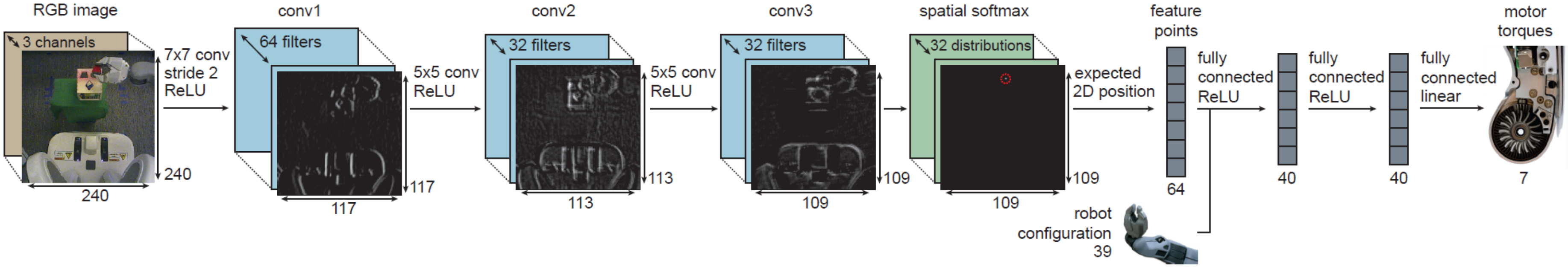
Learning on PR2

[L. et al. ICRA '15]

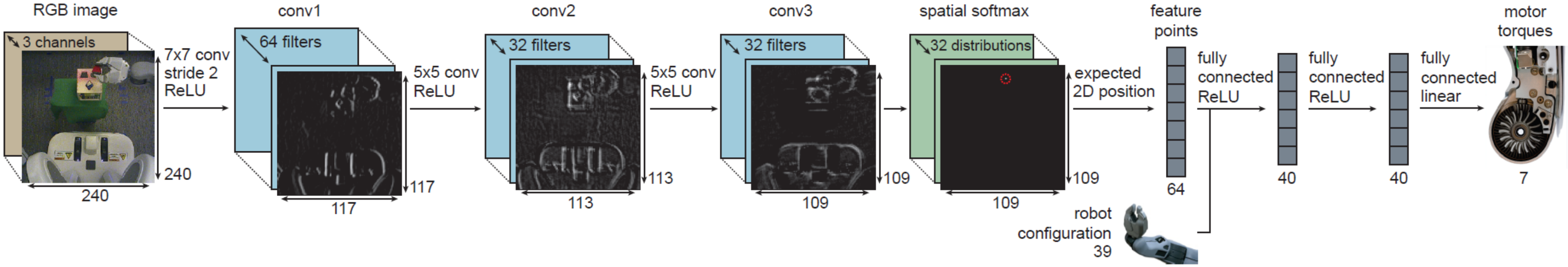
Learning on PR2



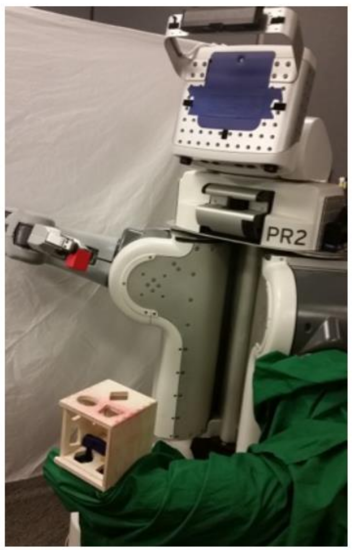
[L. et al. ICRA '15]



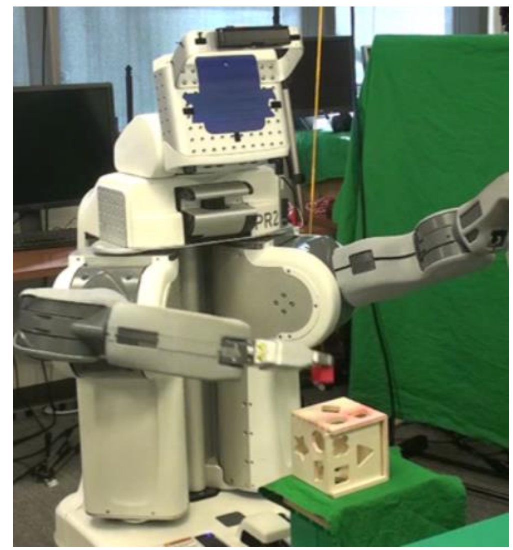
L.*, Finn*, Darrell, Abbeel '15



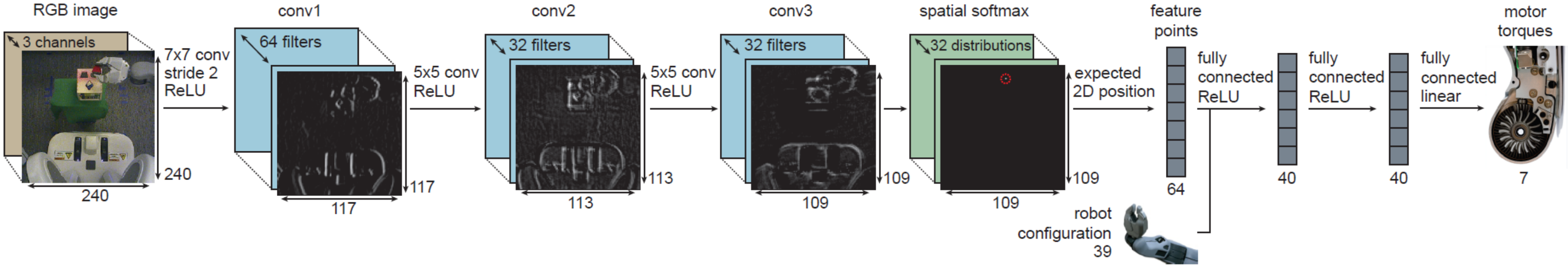
training time



test time

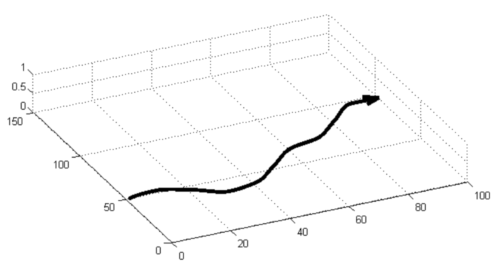


L.*, Finn*, Darrell, Abbeel '15

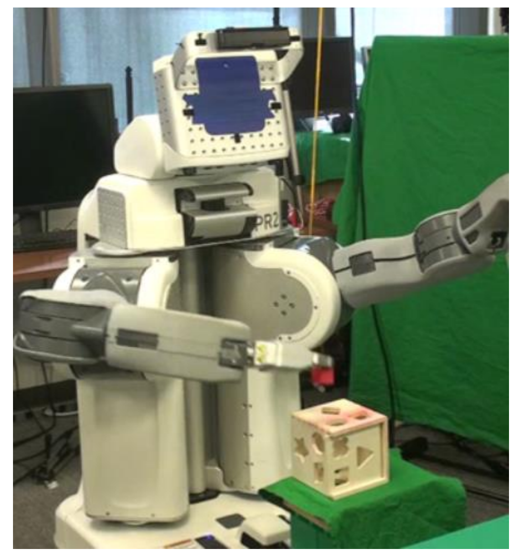


training time

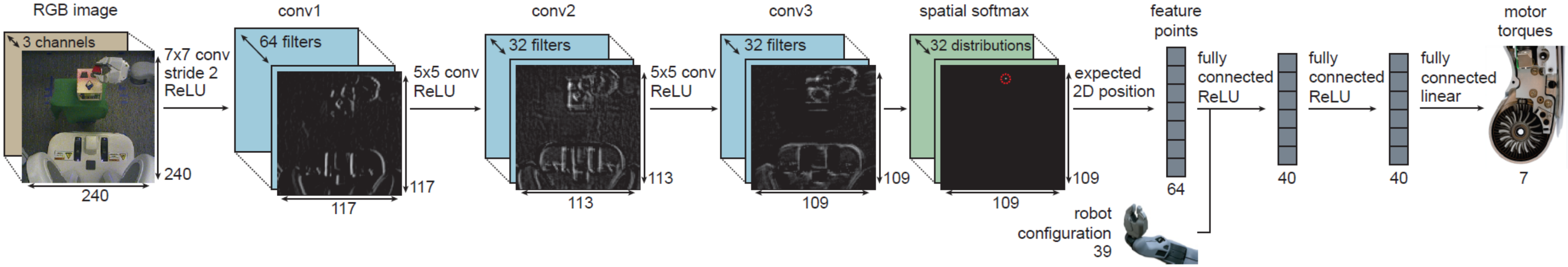
$$\mathbf{x}_t \rightarrow \mathbf{u}_t$$



test time

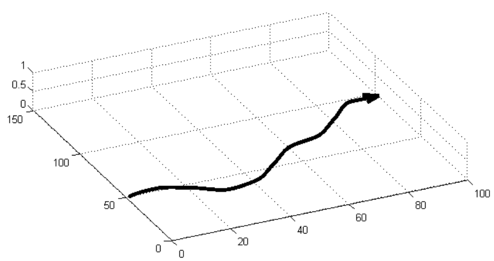


L.*, Finn*, Darrell, Abbeel '15

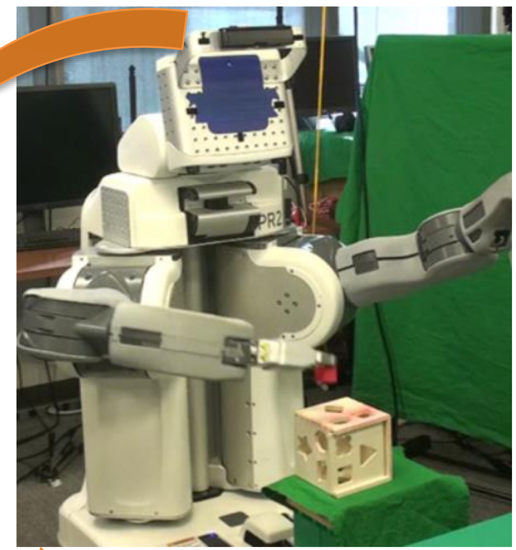


training time

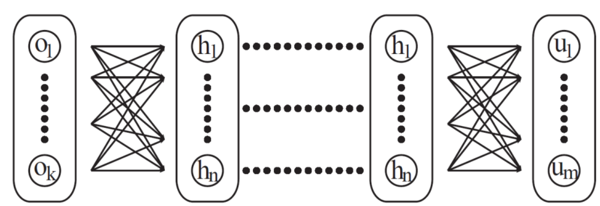
$$\mathbf{x}_t \rightarrow \mathbf{u}_t$$



test time



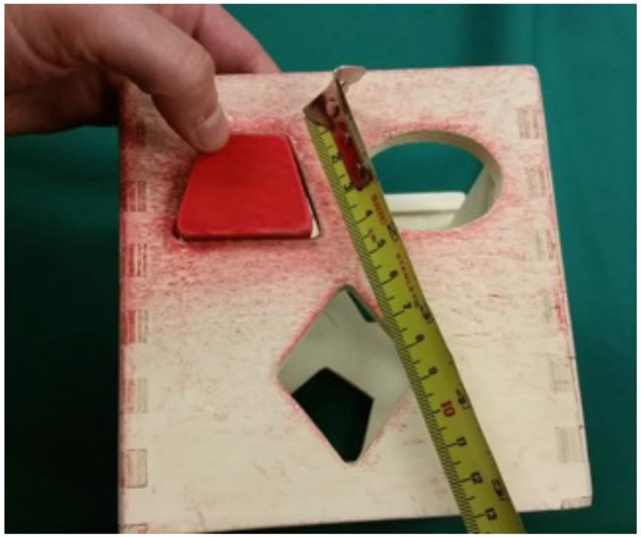
$$\mathbf{o}_t \rightarrow \mathbf{u}_t$$



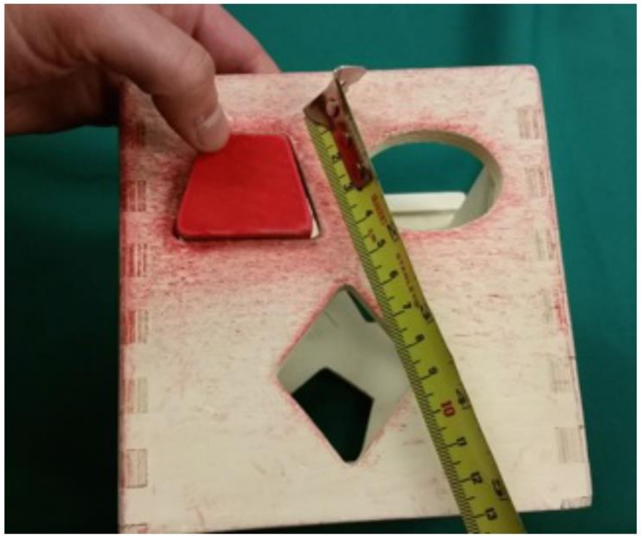
L.*, Finn*, Darrell, Abbeel '15

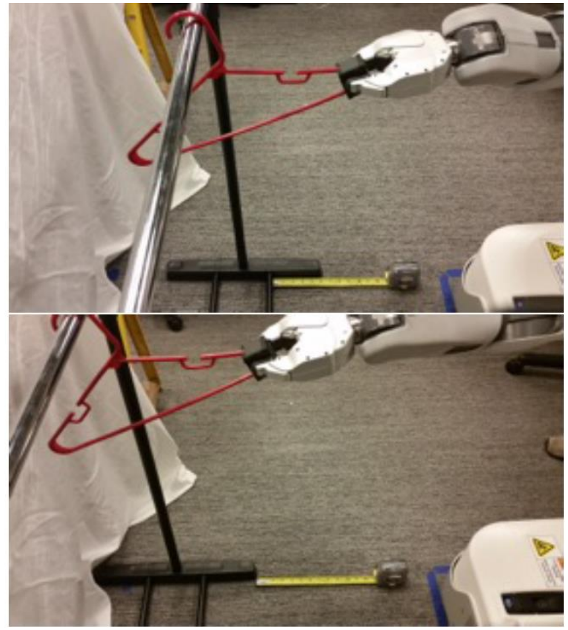
Experimental Tasks

Experimental Tasks

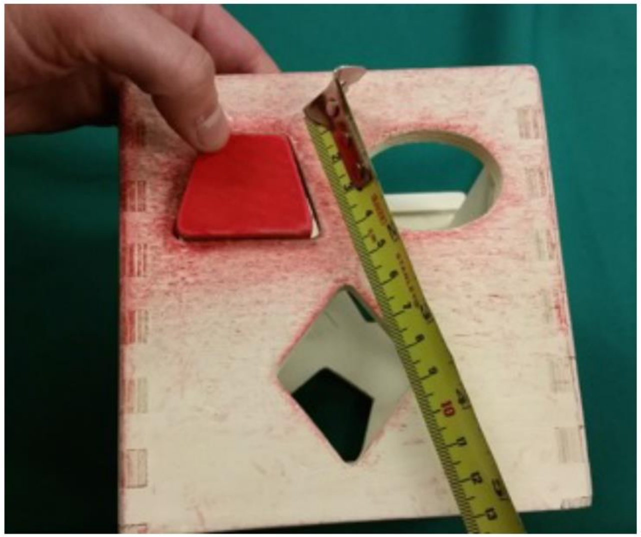


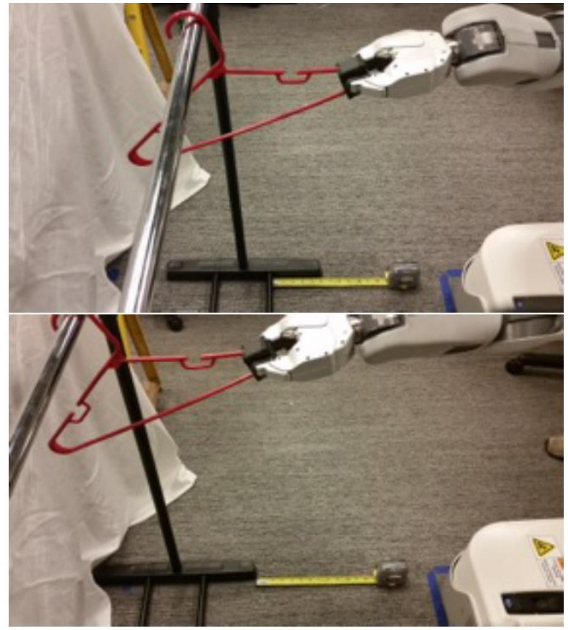
Experimental Tasks



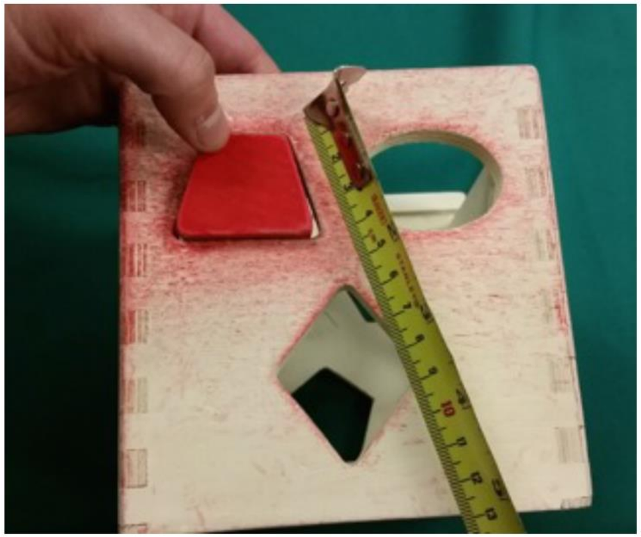


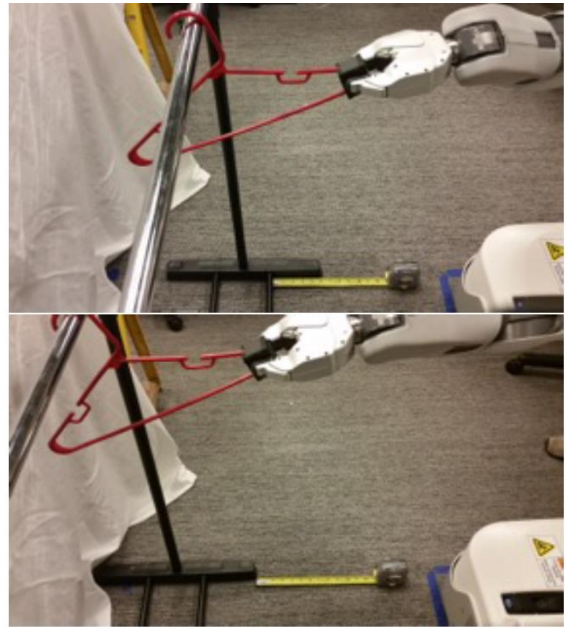
Experimental Tasks





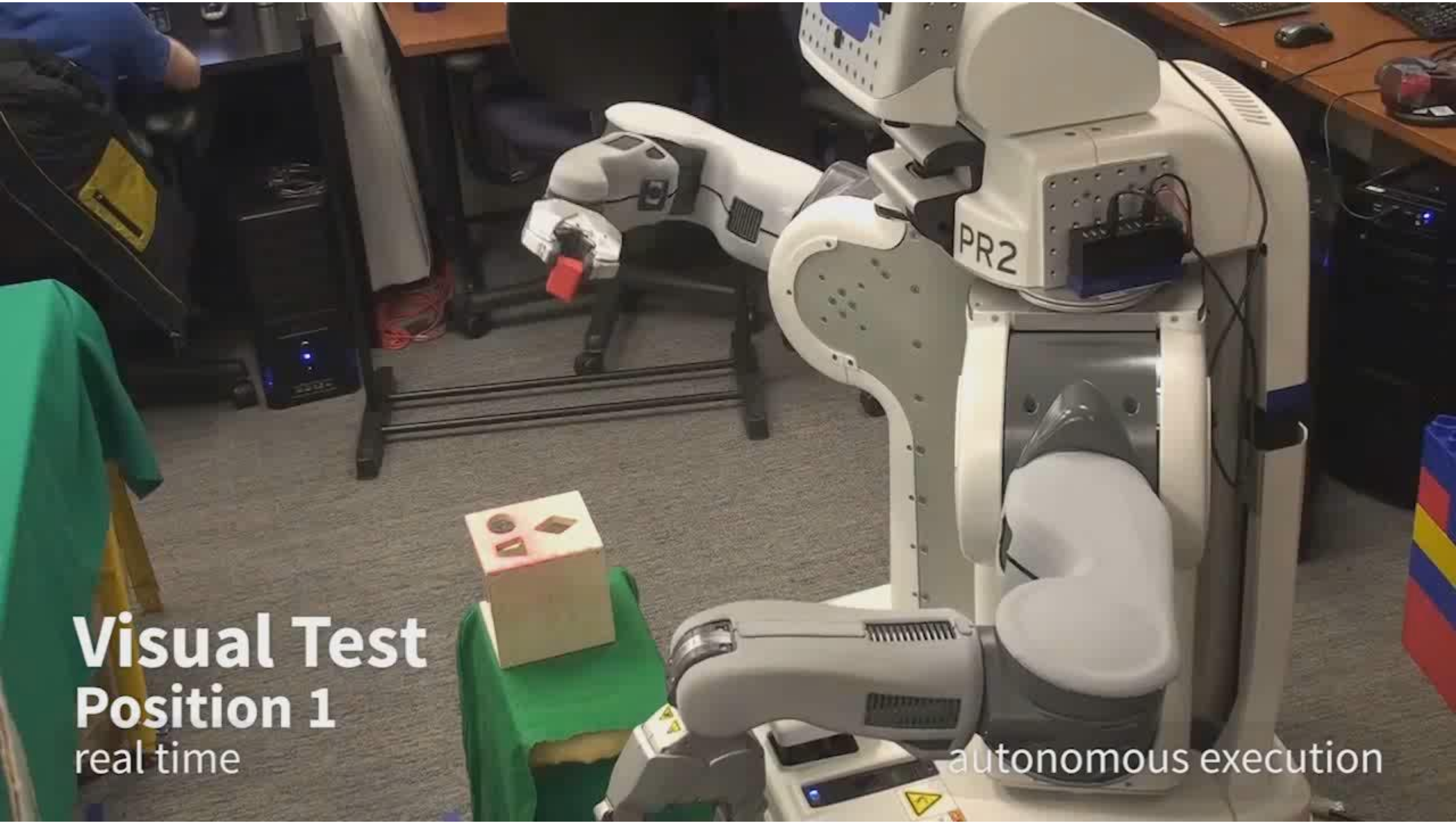
Experimental Tasks



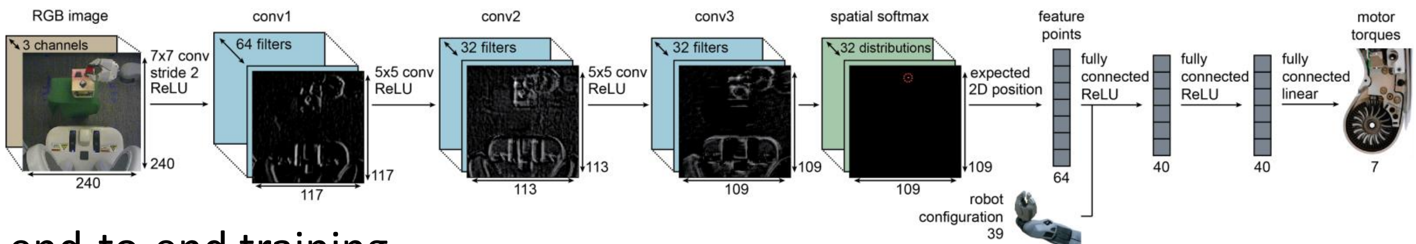


Experimental Tasks

Generalization Experiments

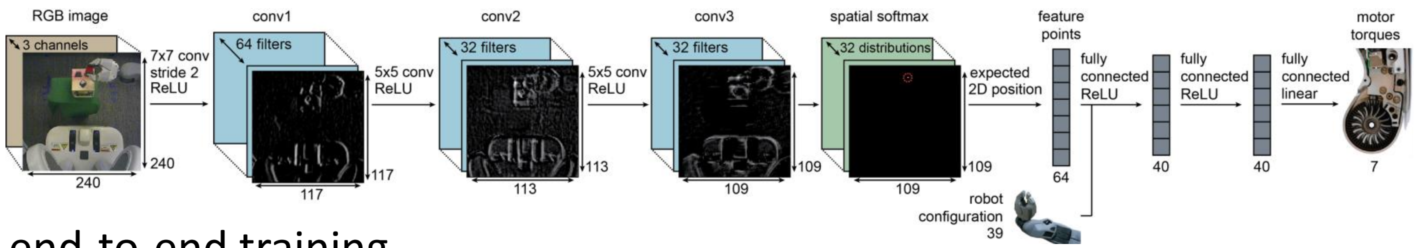


Comparisons

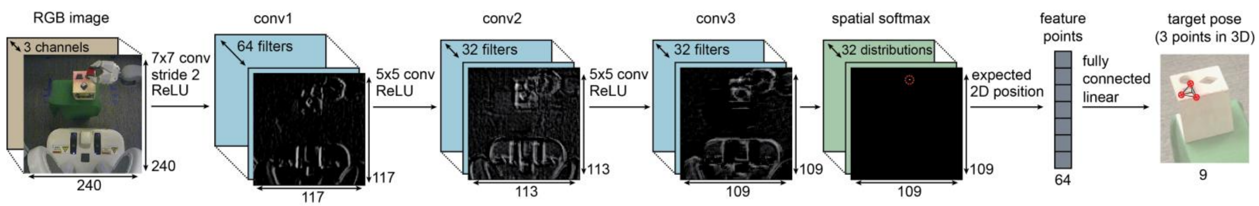


end-to-end training

Comparisons

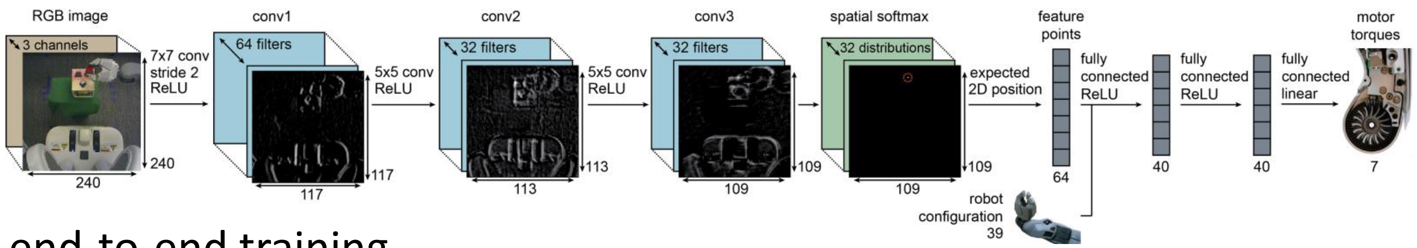


end-to-end training

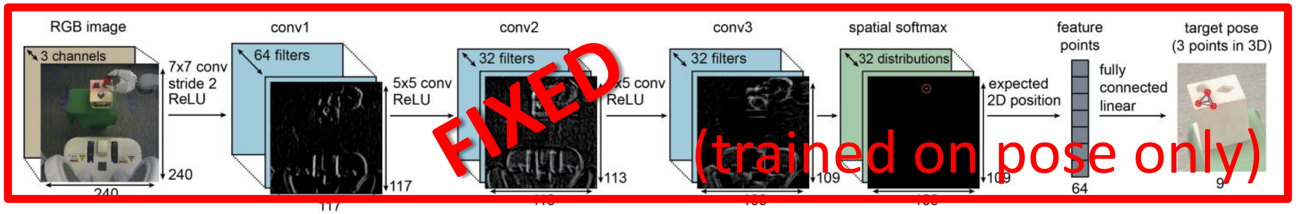


pose prediction

Comparisons

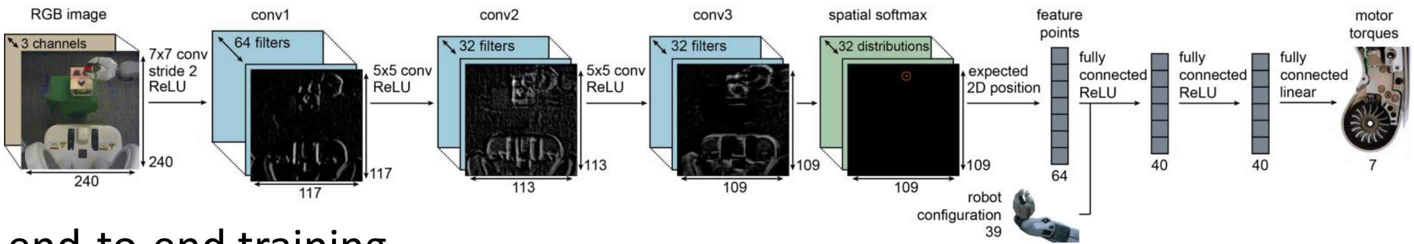


end-to-end training

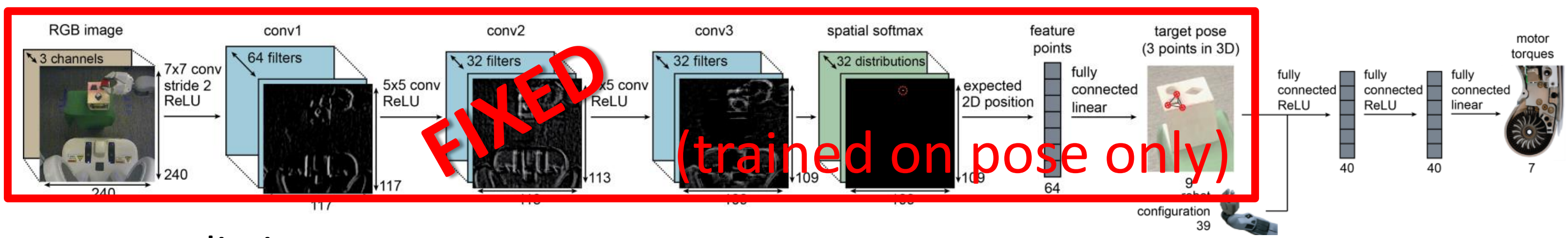


pose prediction

Comparisons

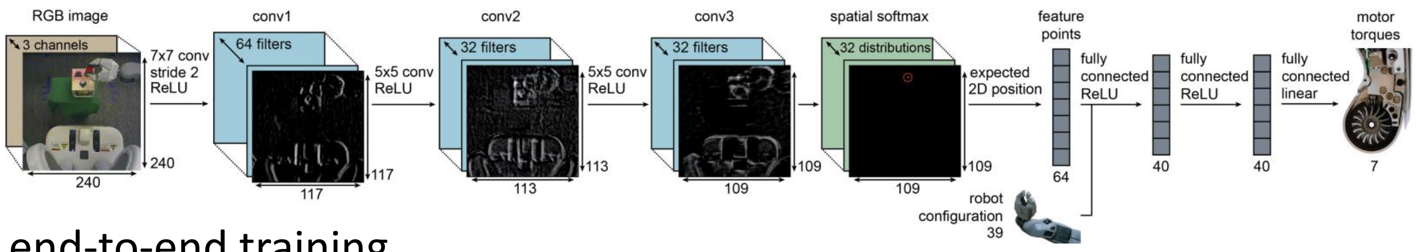


end-to-end training

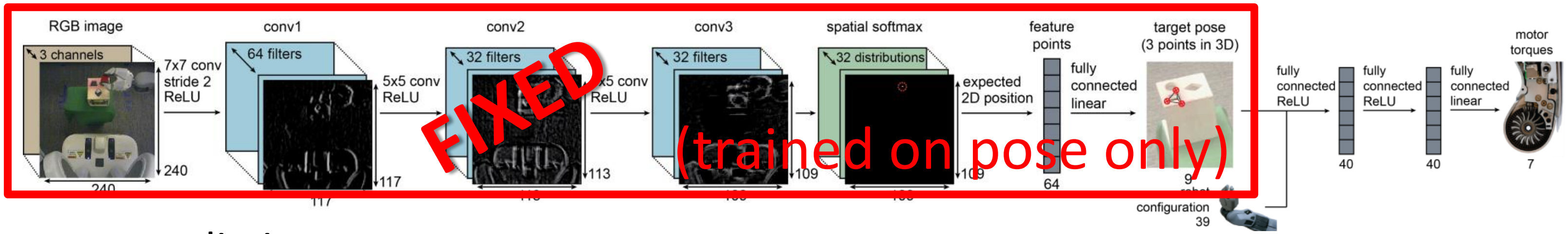


pose prediction

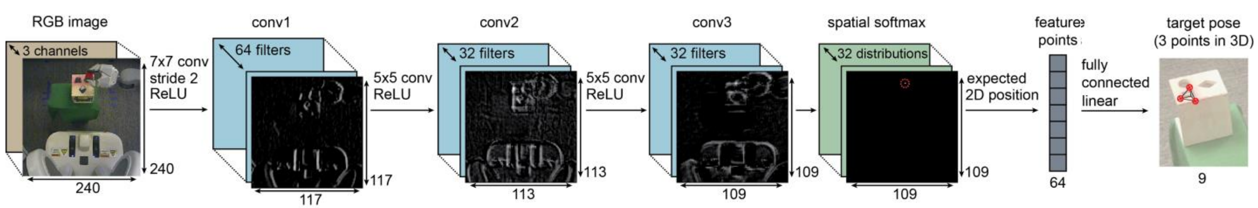
Comparisons



end-to-end training

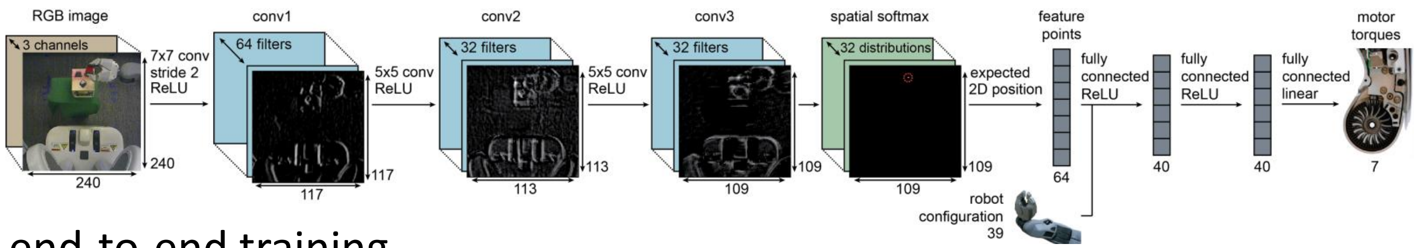


pose prediction

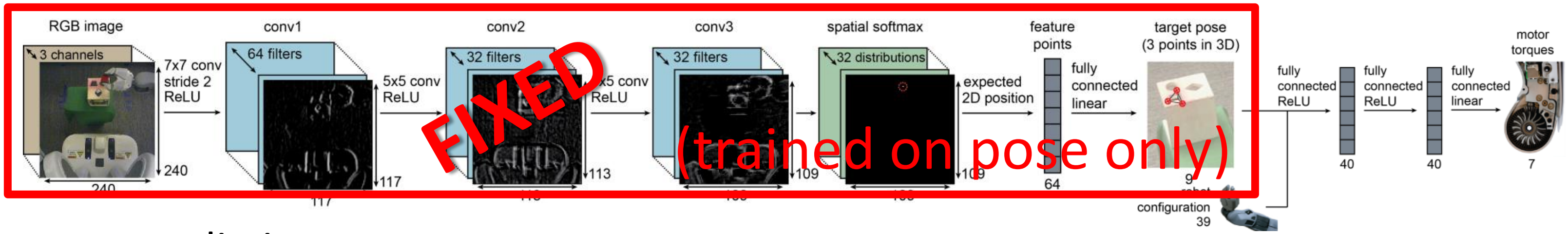


pose features

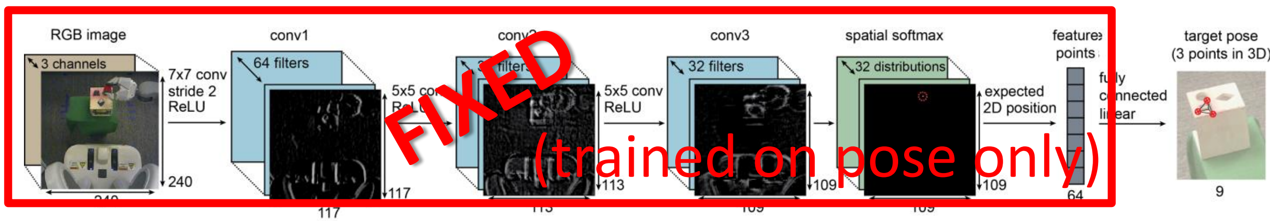
Comparisons



end-to-end training

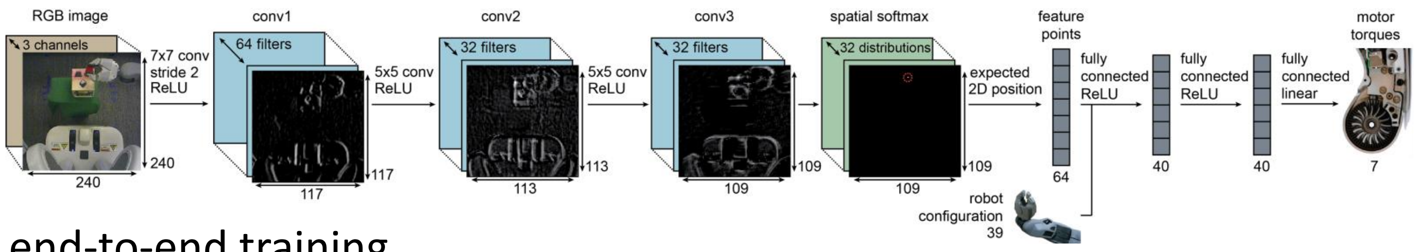


pose prediction

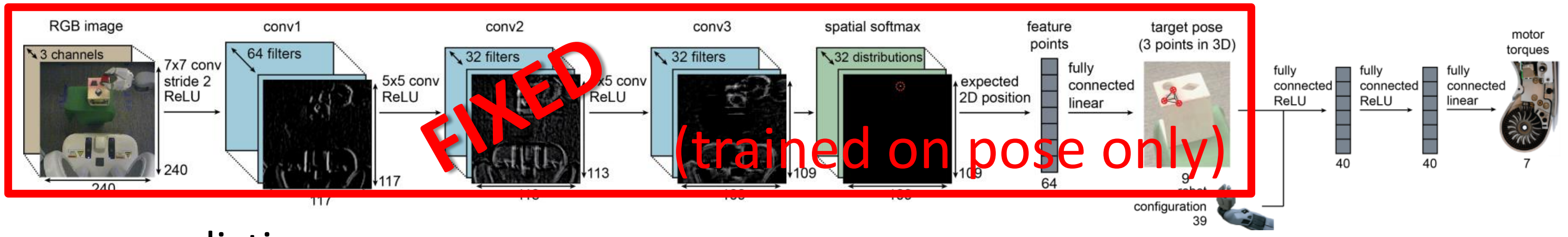


pose features

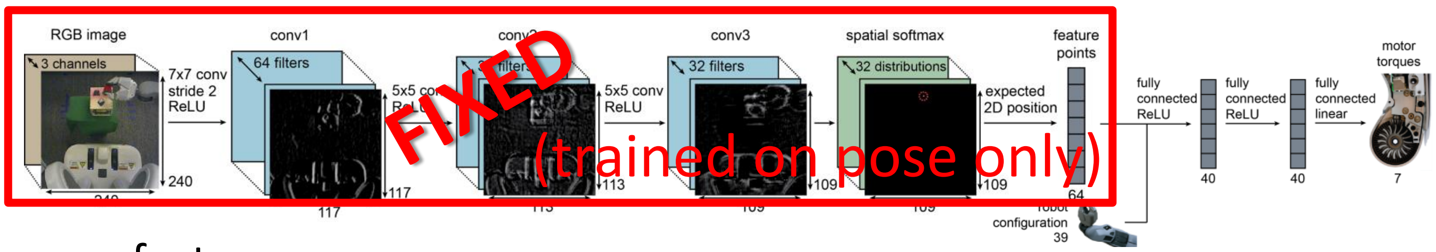
Comparisons



end-to-end training



pose prediction



pose features

Comparisons

Comparisons

coat hanger	success rate
pose prediction	55.6%

shape sorting cube	success rate
pose prediction	0%

toy claw hammer	success rate
pose prediction	8.9%

bottle cap	success rate
pose prediction	n/a

Comparisons

coat hanger	success rate
pose prediction	55.6%
pose features	88.9%

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%

bottle cap	success rate
pose prediction	n/a
pose features	55.6%

Comparisons

coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

Comparisons

coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\mathbf{1.30 \pm 0.73}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73

Comparisons

coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\boxed{\mathbf{1.30 \pm 0.73}}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73

Comparisons

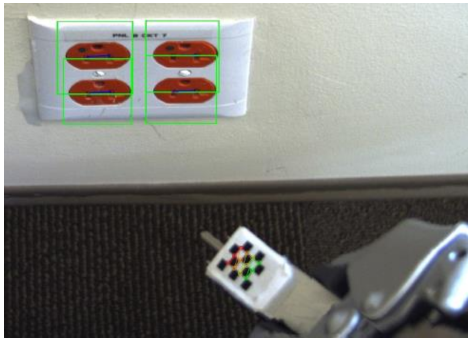
coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\boxed{\mathbf{1.30 \pm 0.73}}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73



Meeussen et al. (Willow Garage)

Comparisons

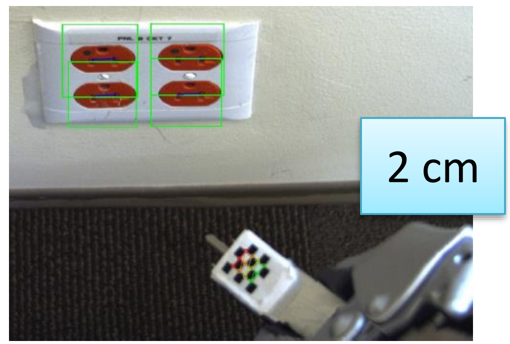
coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\mathbf{1.30 \pm 0.73}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73



Meeussen et al. (Willow Garage)

Comparisons

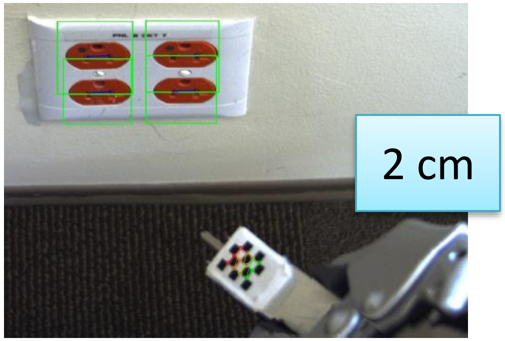
coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

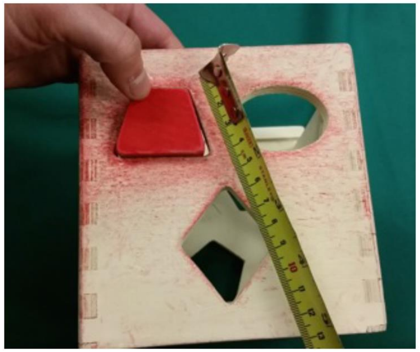
toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\boxed{\mathbf{1.30 \pm 0.73}}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73



Meeussen et al. (Willow Garage)



Comparisons

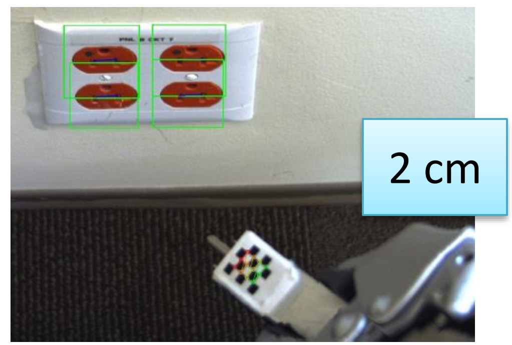
coat hanger	success rate
pose prediction	55.6%
pose features	88.9%
end-to-end training	**100%**

shape sorting cube	success rate
pose prediction	0%
pose features	70.4%
end-to-end training	**96.3%**

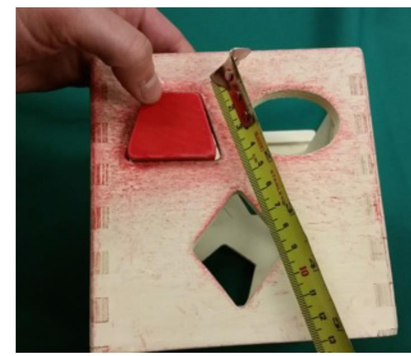
toy claw hammer	success rate
pose prediction	8.9%
pose features	62.2%
end-to-end training	**91.1%**

bottle cap	success rate
pose prediction	n/a
pose features	55.6%
end-to-end training	**88.9%**

network architecture	test error (cm)
softmax + feature points (**ours**)	$\boxed{\mathbf{1.30} \pm \mathbf{0.73}}$
softmax + fully connected layer	2.59 ± 1.19
fully connected layer	4.75 ± 2.29
max-pooling + fully connected	3.71 ± 1.73



Meeussen et al. (Willow Garage)



Guided Policy Search Applications

Guided Policy Search Applications

manipulation

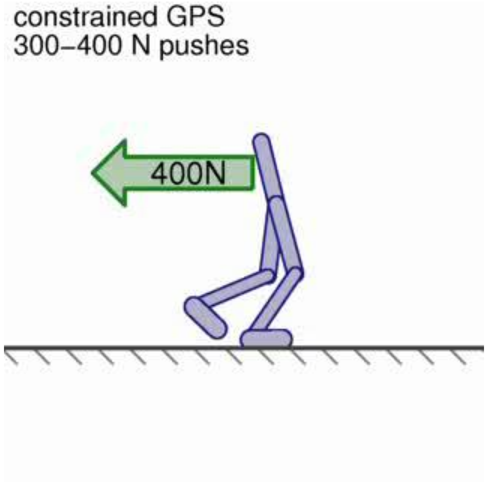
with N. Wagener and P. Abbeel

Guided Policy Search Applications

manipulation

with N. Wagener and P. Abbeel

locomotion



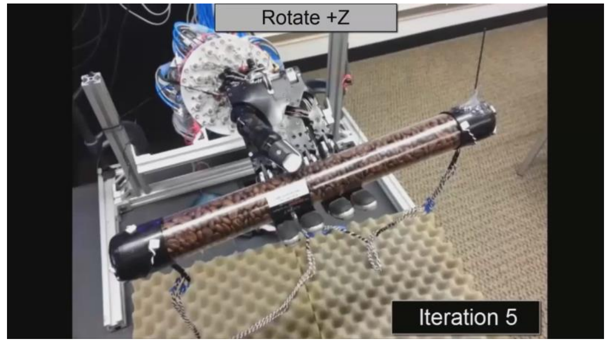
with V. Koltun

Guided Policy Search Applications

manipulation

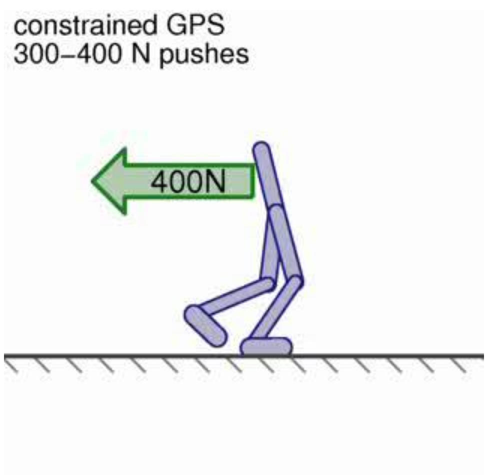
with N. Wagener and P. Abbeel

dexterous hands



with V. Kumar and E. Todorov

locomotion



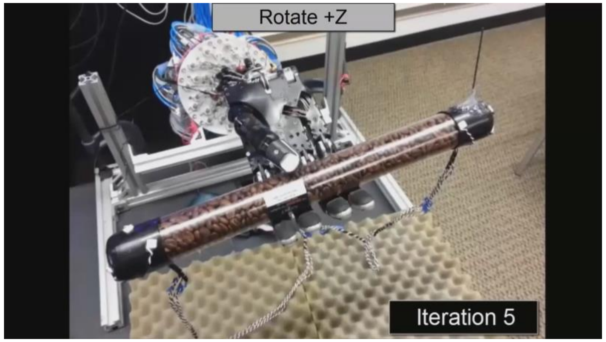
with V. Koltun

Guided Policy Search Applications

manipulation

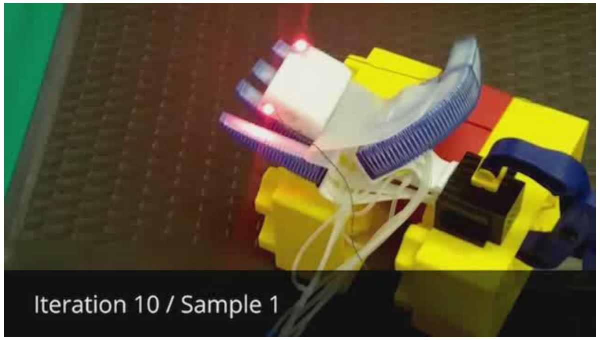
with N. Wagener and P. Abbeel

dexterous hands



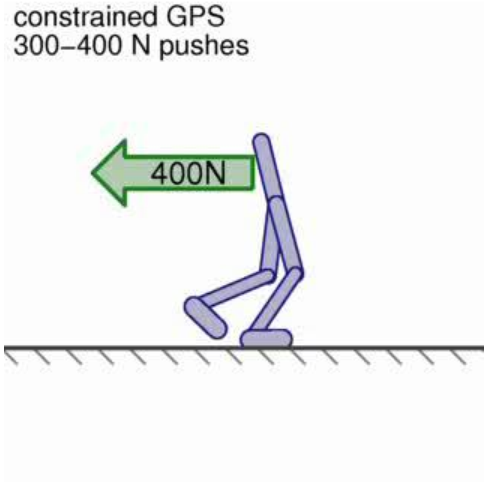
with V. Kumar and E. Todorov

soft hands



with A. Gupta, C. Eppner, P. Abbeel

locomotion



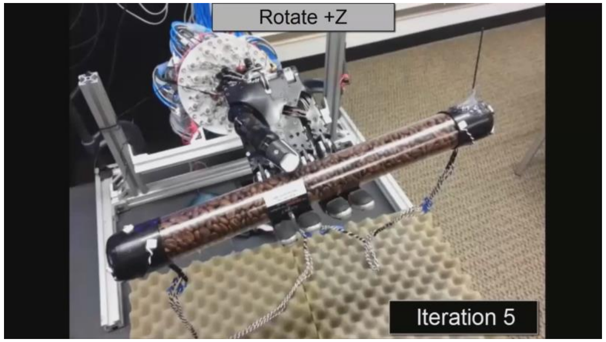
with V. Koltun

Guided Policy Search Applications

manipulation

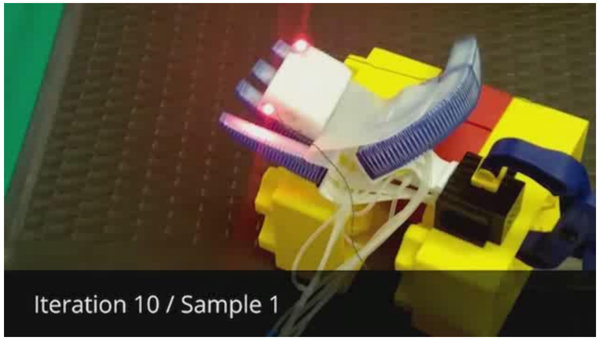
with N. Wagener and P. Abbeel

dexterous hands



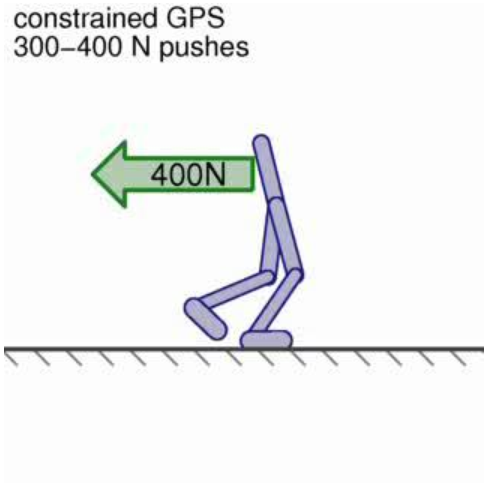
with V. Kumar and E. Todorov

soft hands



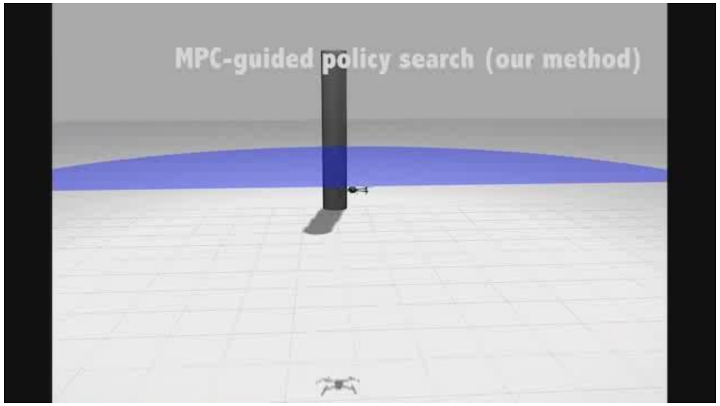
with A. Gupta, C. Eppner, P. Abbeel

locomotion



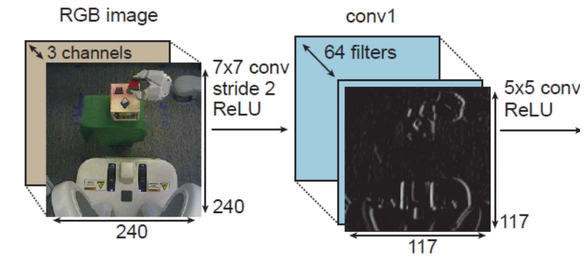
with V. Koltun

aerial vehicles



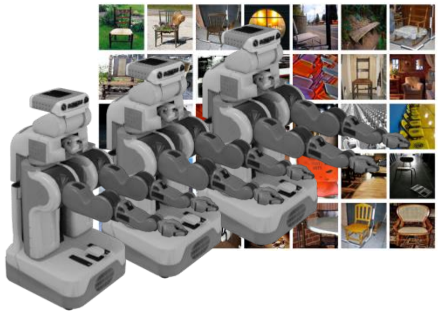
with G. Kahn, T. Zhang, P. Abbeel

Overview



Training visuomotor policies

Deep robotic learning at scale



Future directions

ingredients for success in learning:

supervised learning:

ingredients for success in learning:

supervised learning:

- ✓ computation

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

learning sensorimotor skills:

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

learning sensorimotor skills:

- ✓ computation

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

learning sensorimotor skills:

- ✓ computation
- ~ algorithms

ingredients for success in learning:

supervised learning:	learning sensorimotor skills:
✓ computation	✓ computation
✓ algorithms	~ algorithms
✓ data	? data

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

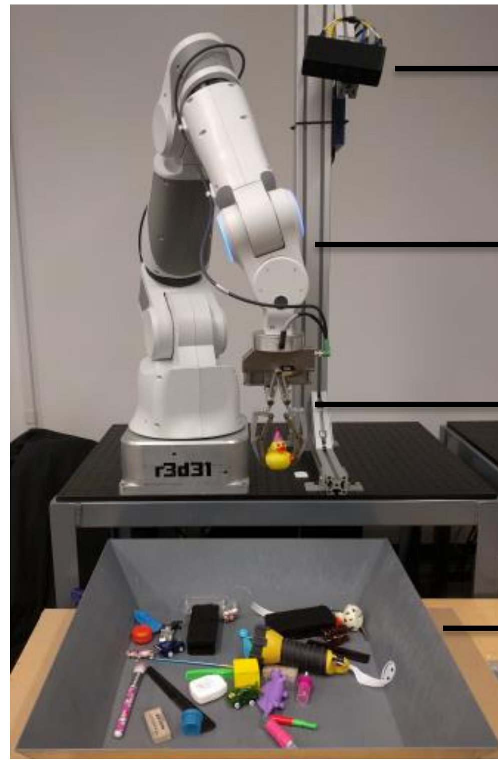
learning sensorimotor skills:

- ✓ computation
- ~ algorithms
- ? data

L., Pastor, Krizhevsky, Quillen ‘16

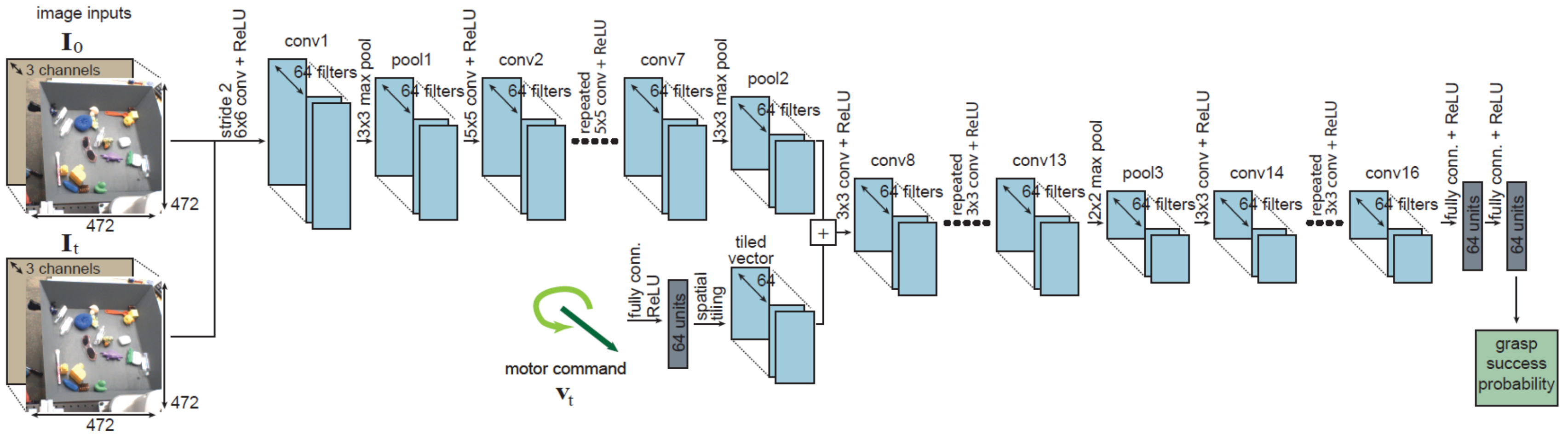
Grasping with Learned Hand-Eye Coordination

- 800,000 grasp attempts for training (3,000 robot-hours)
- monocular camera (no depth)
- 2-5 Hz update
- no prior knowledge



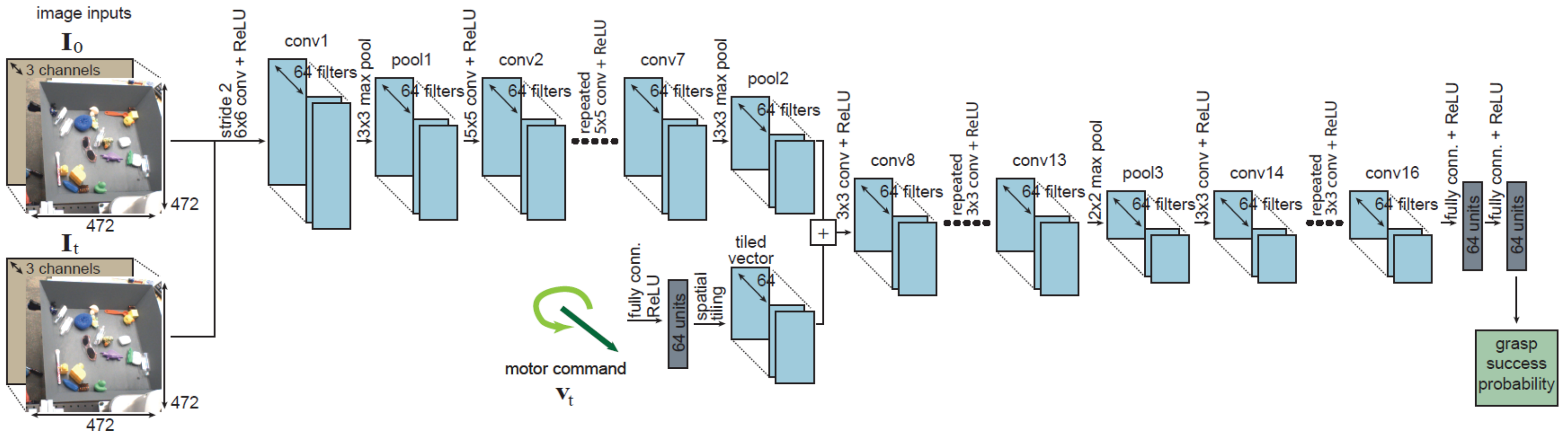
L., Pastor, Krizhevsky, Quillen '16

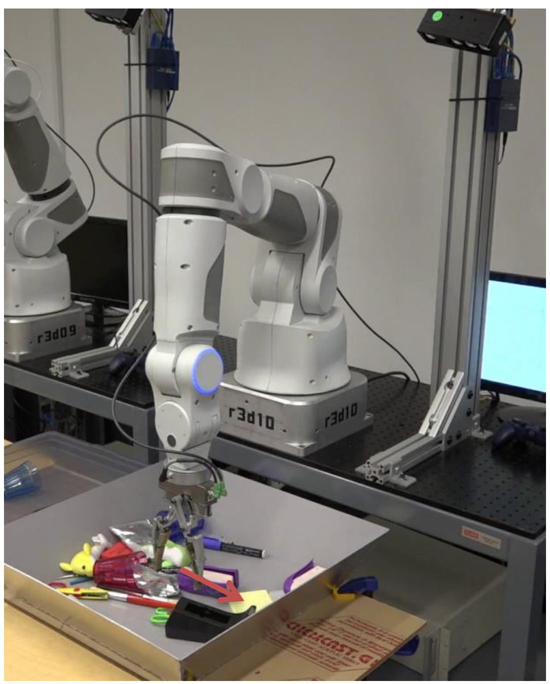
Using Grasp Success Prediction



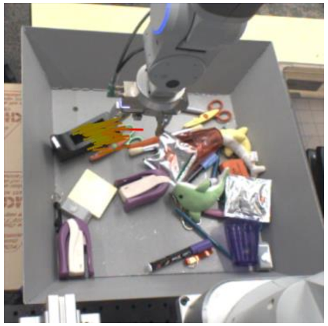
L., Pastor, Krizhevsky, Quillen ‘16

Using Grasp Success Prediction



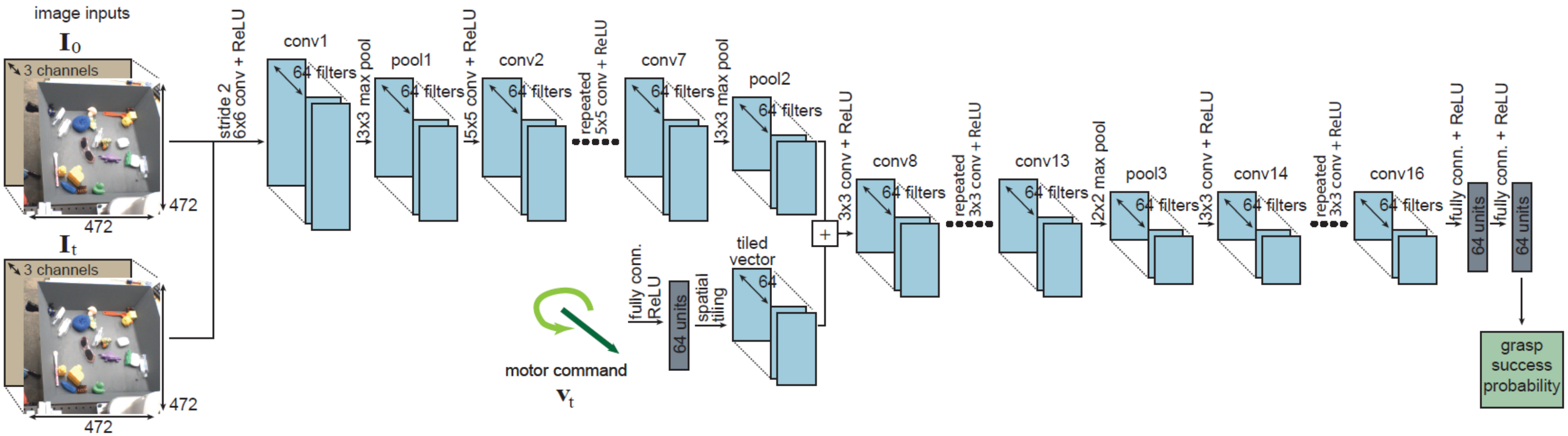


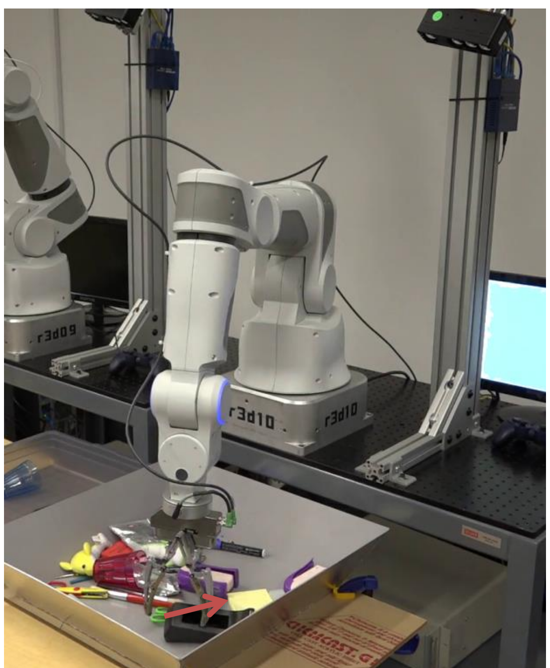
training



L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction

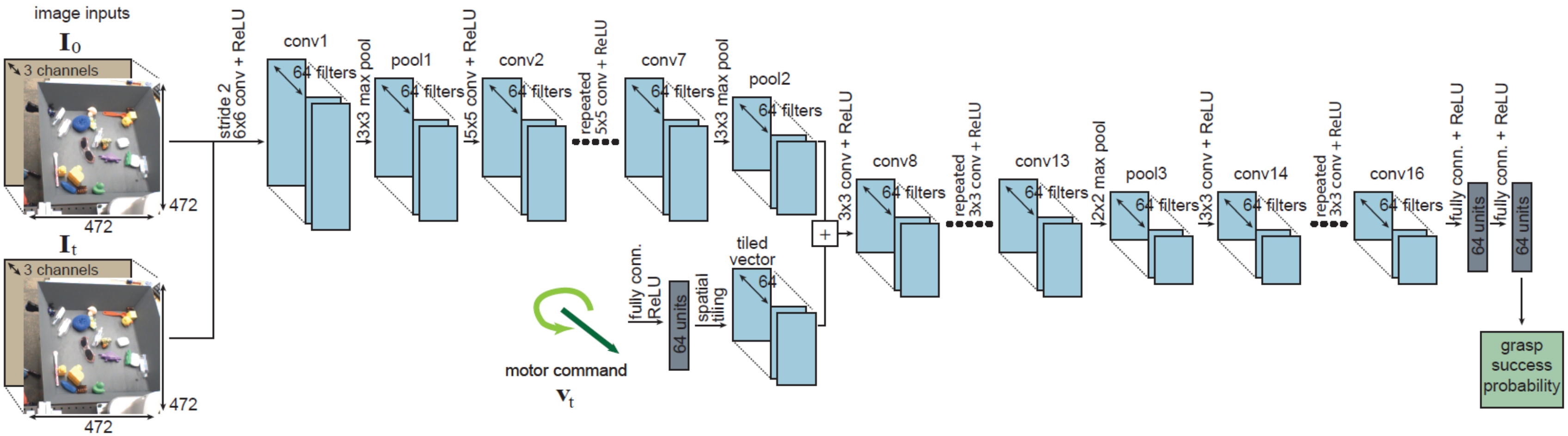


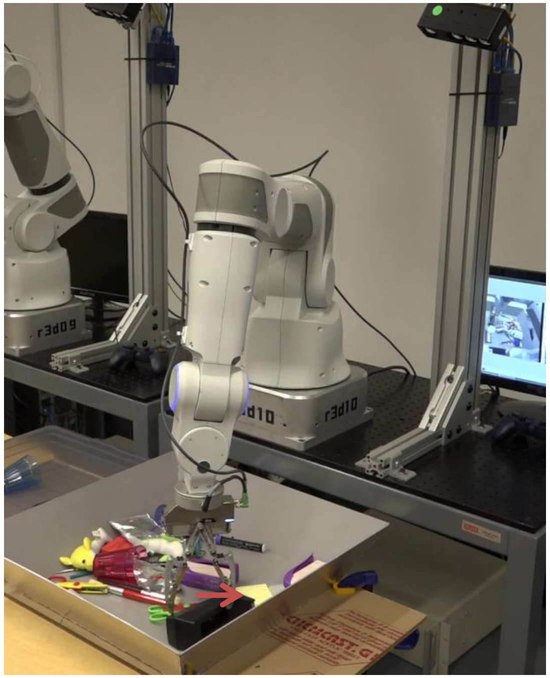


training

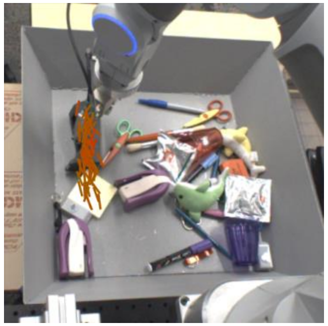
L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction



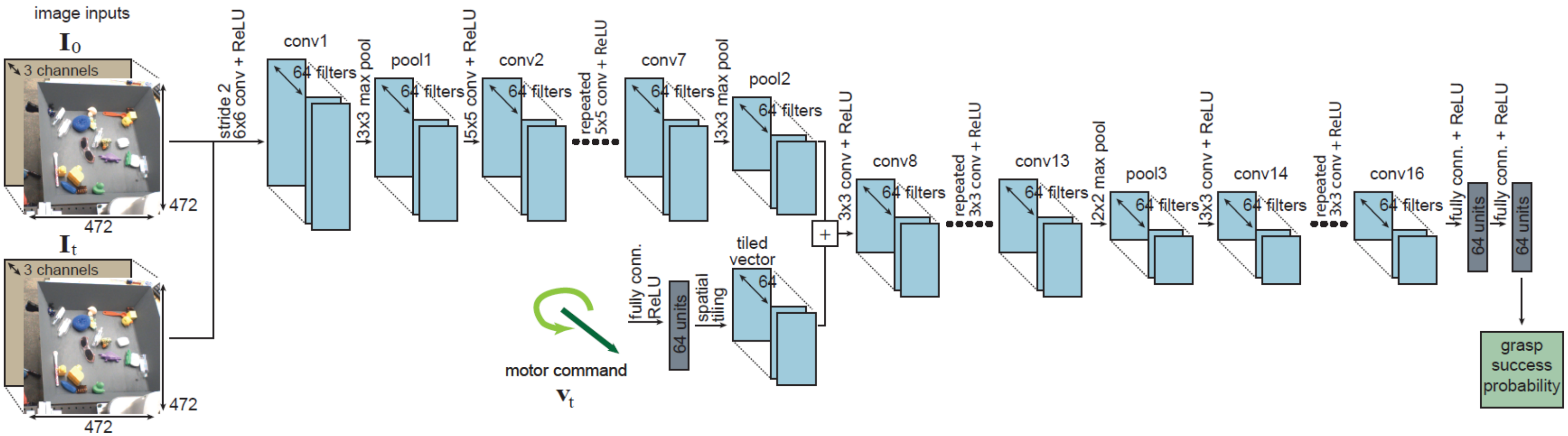


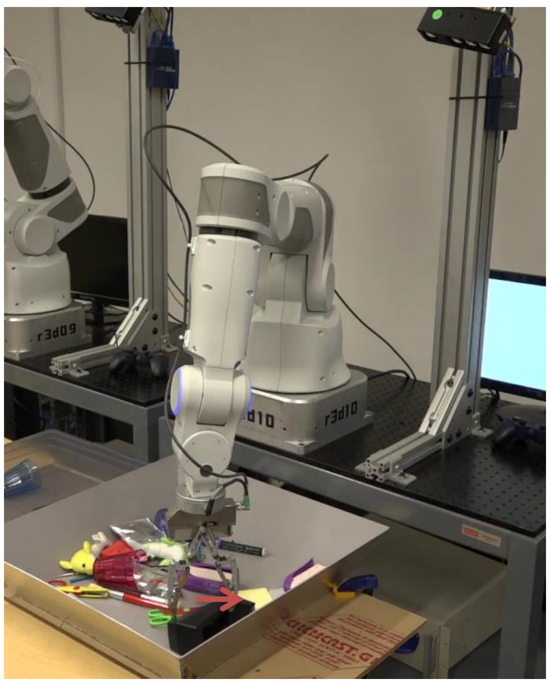
training



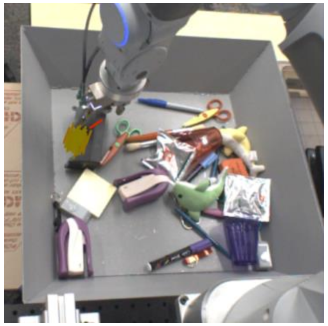
L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction



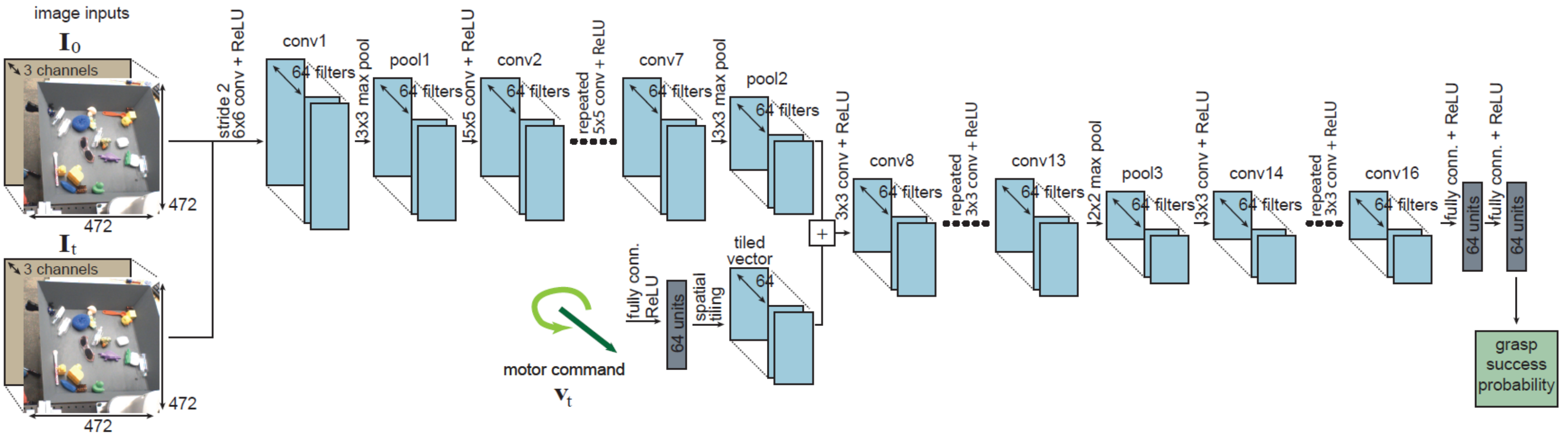


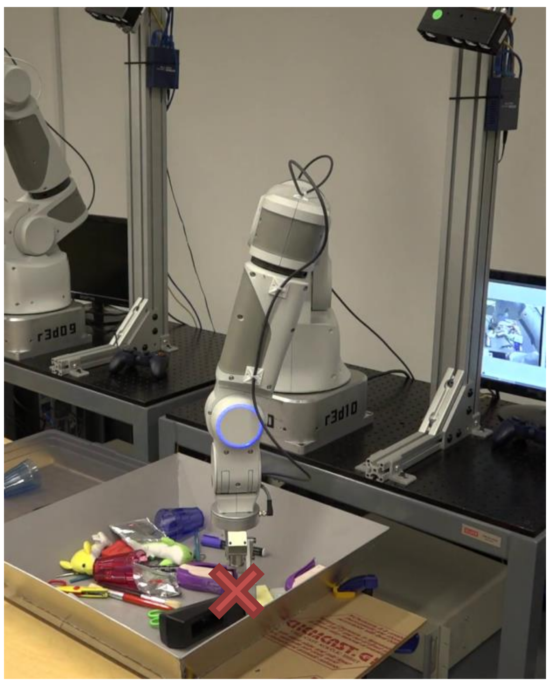
training



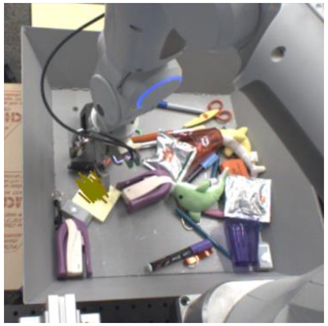
L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction



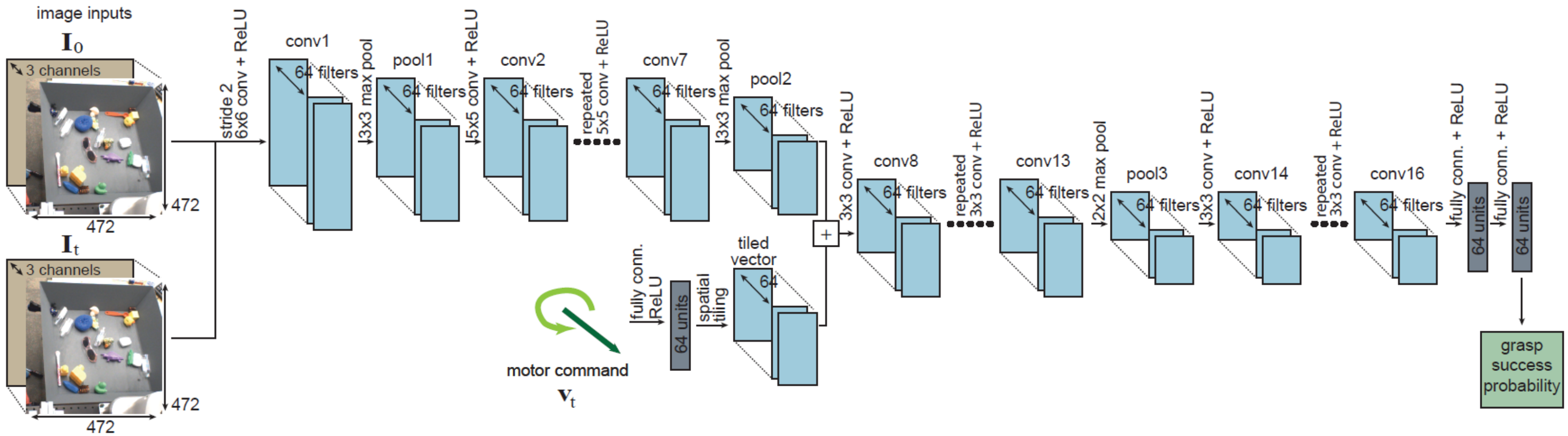


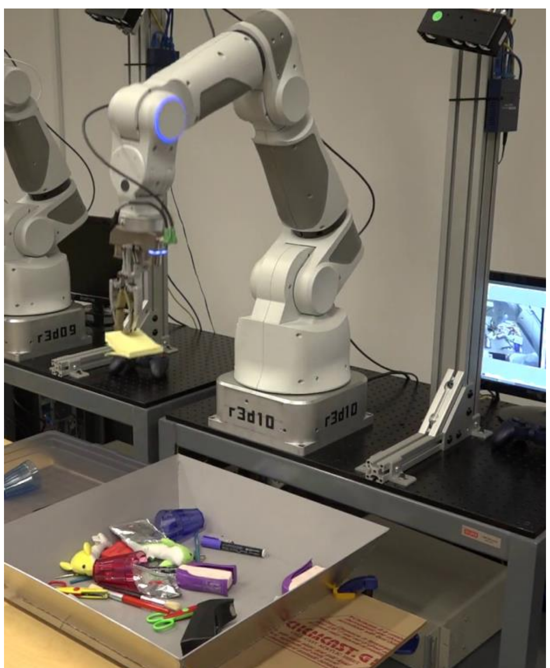
training



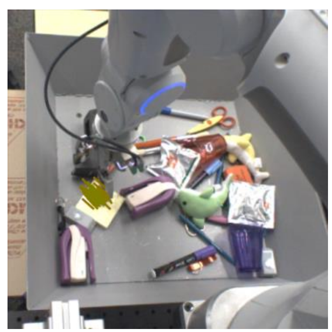
L., Pastor, Krizhevsky, Quillen ‘16

Using Grasp Success Prediction



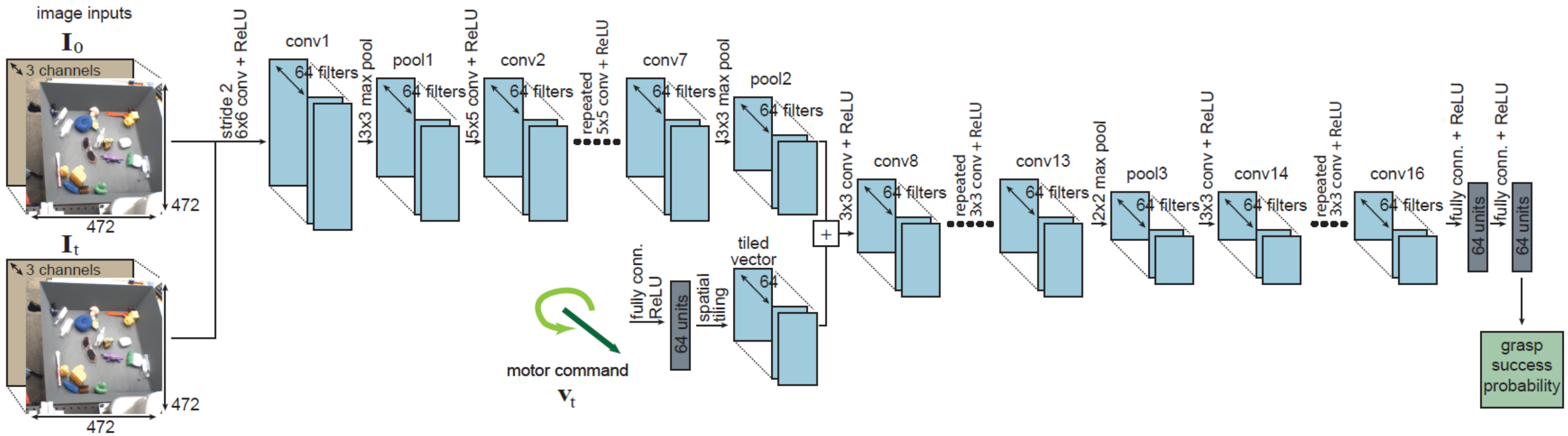


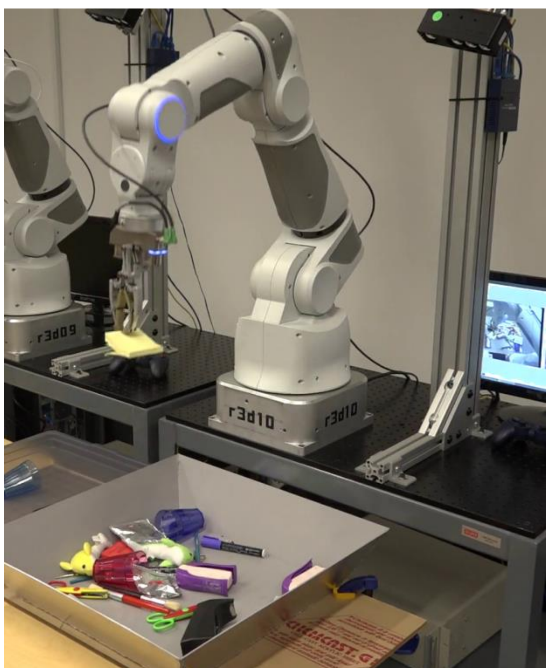
training



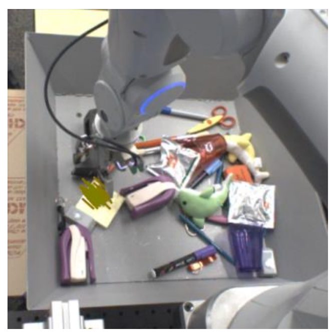
L., Pastor, Krizhevsky, Quillen '16

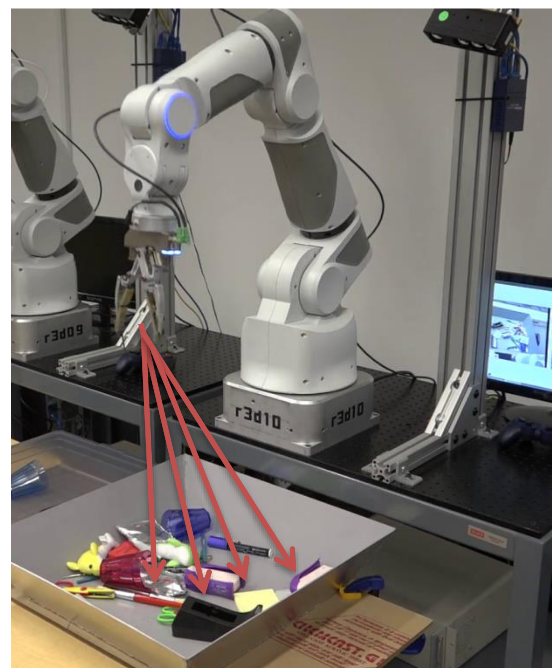
Using Grasp Success Prediction



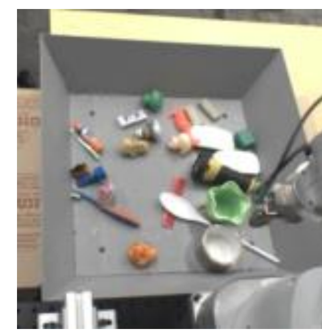


training



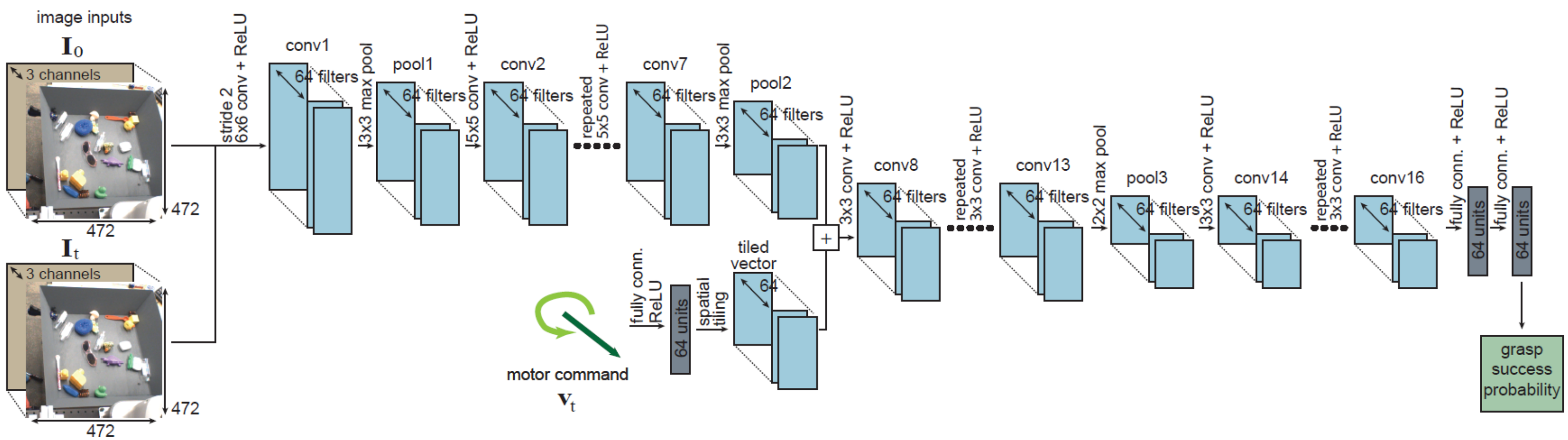


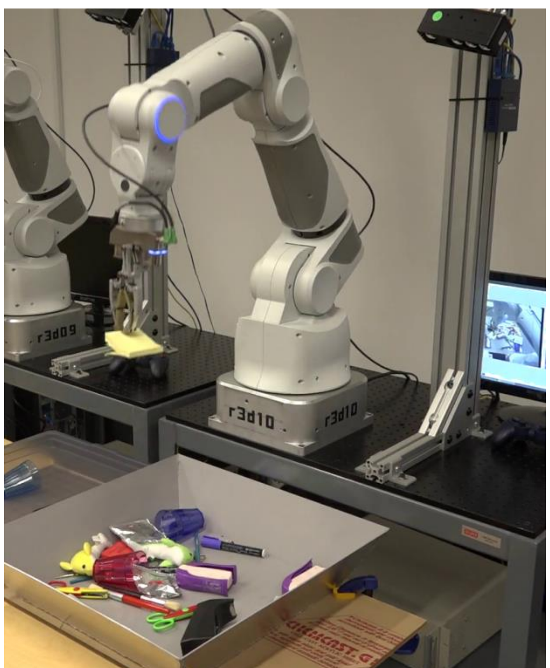
testing



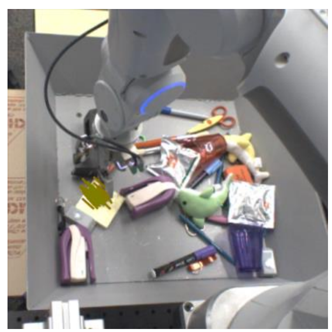
L., Pastor, Krizhevsky, Quillen '16

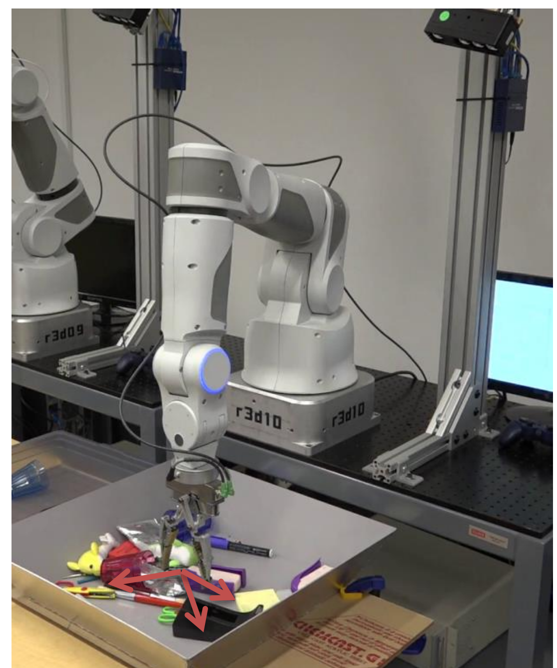
Using Grasp Success Prediction



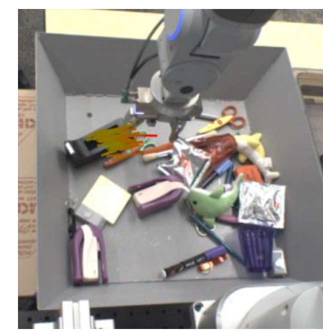


training



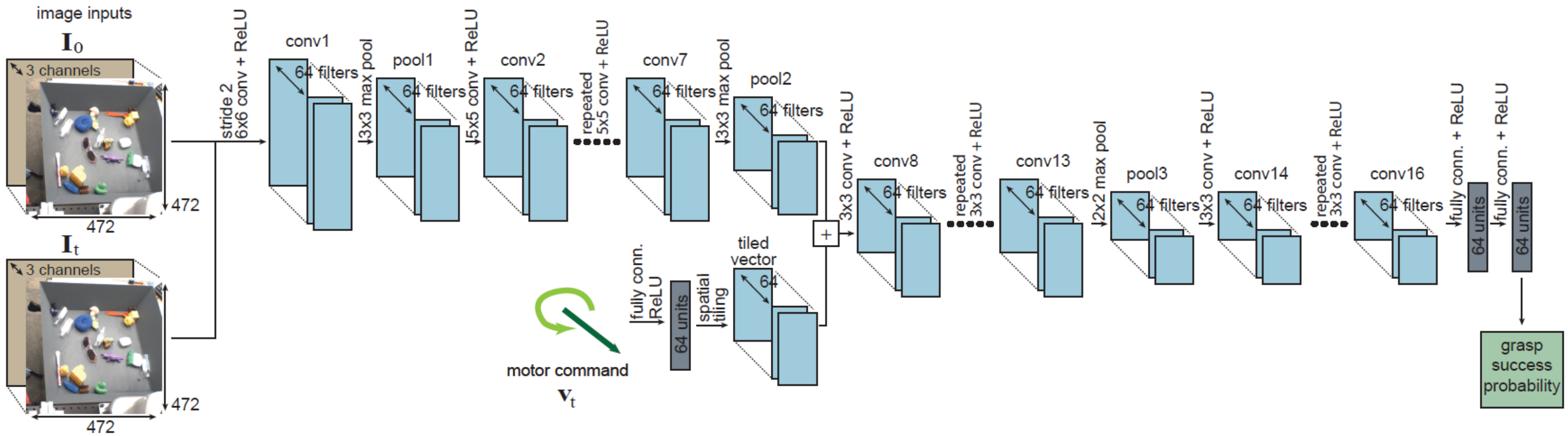


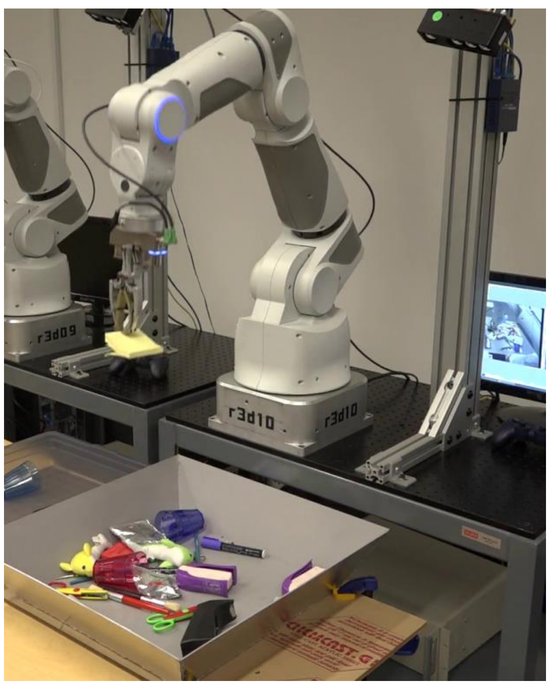
testing



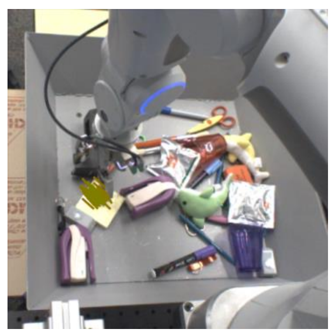
L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction

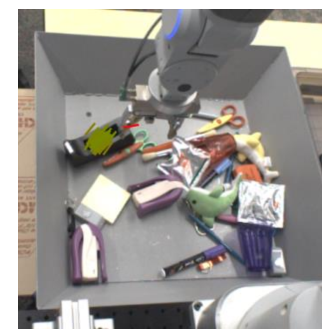




training

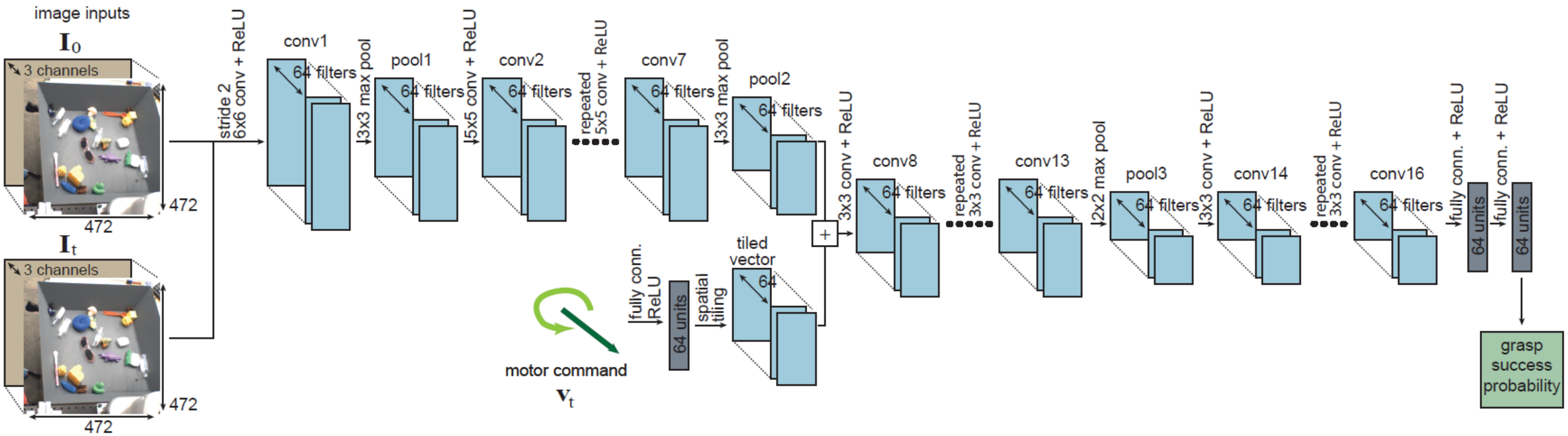


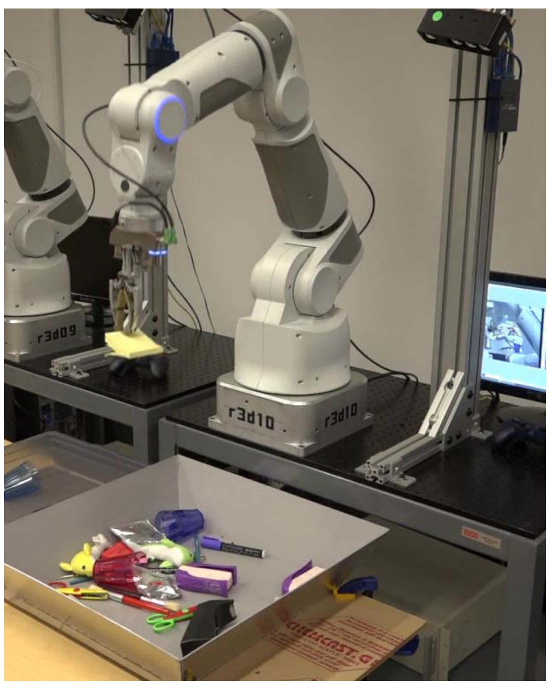
testing



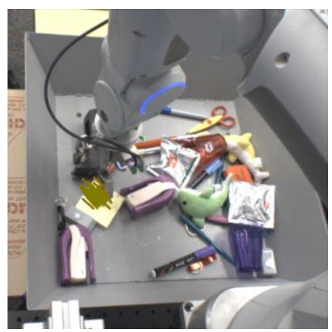
L., Pastor, Krizhevsky, Quillen '16

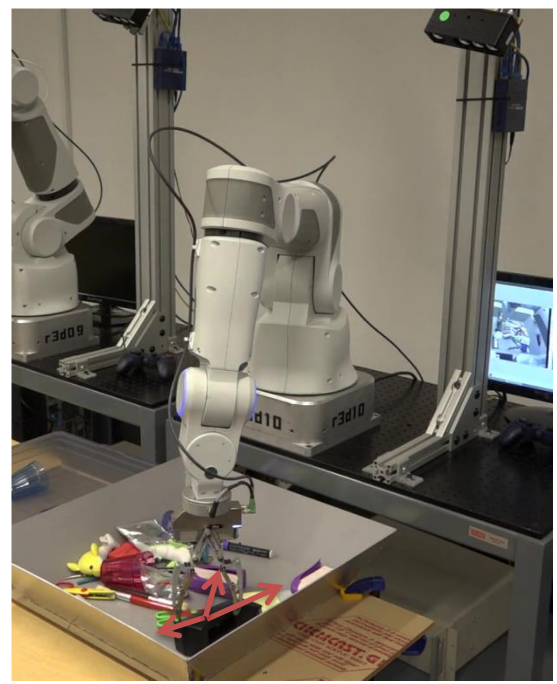
Using Grasp Success Prediction



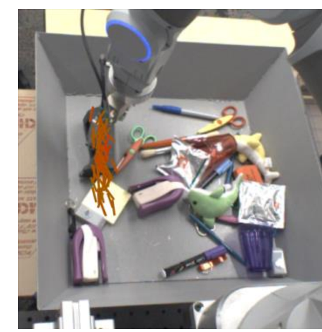


training



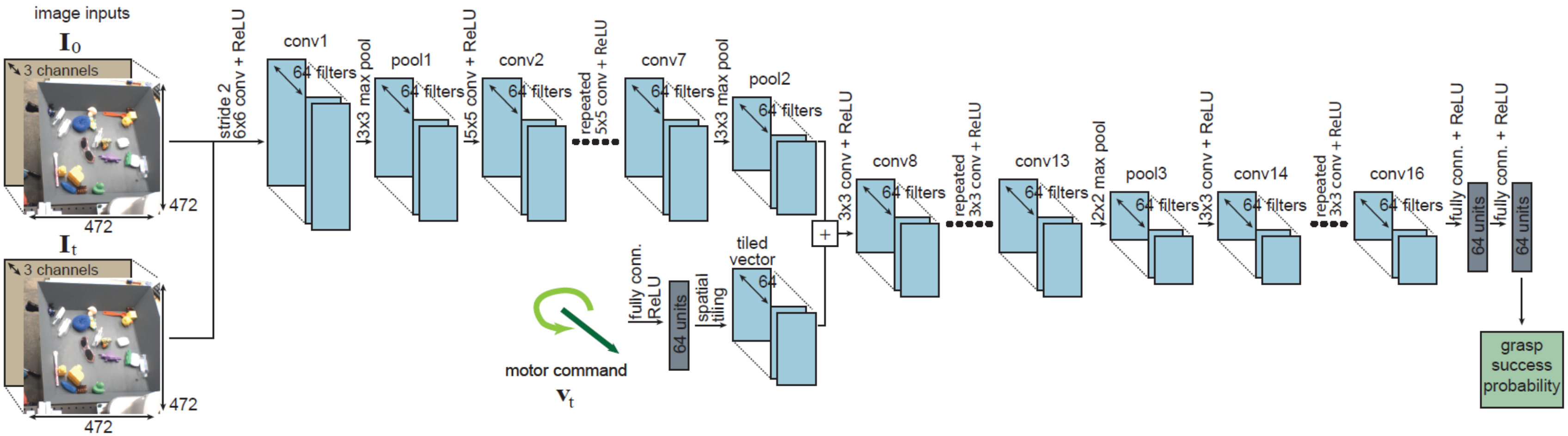


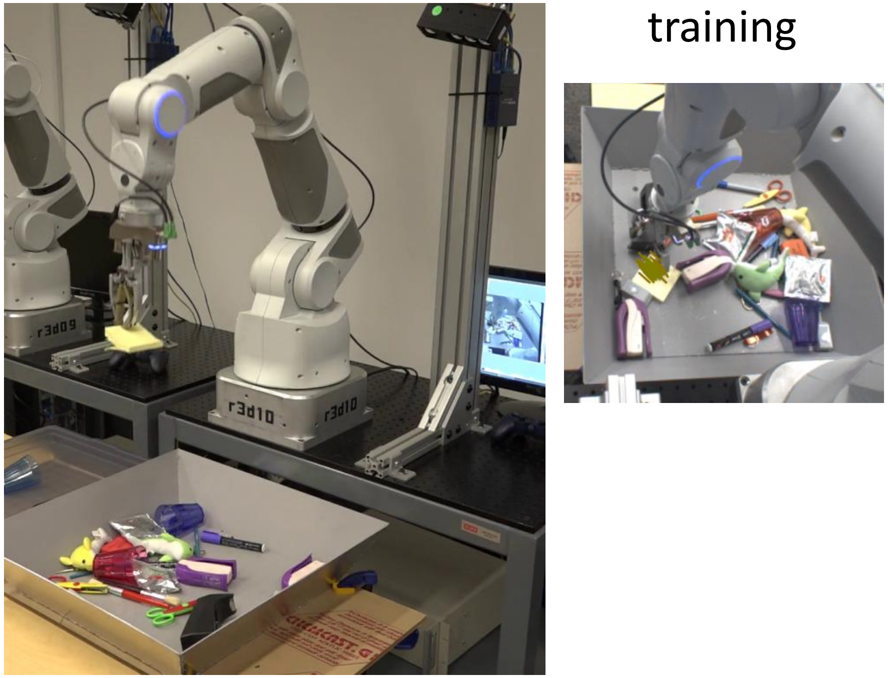
testing



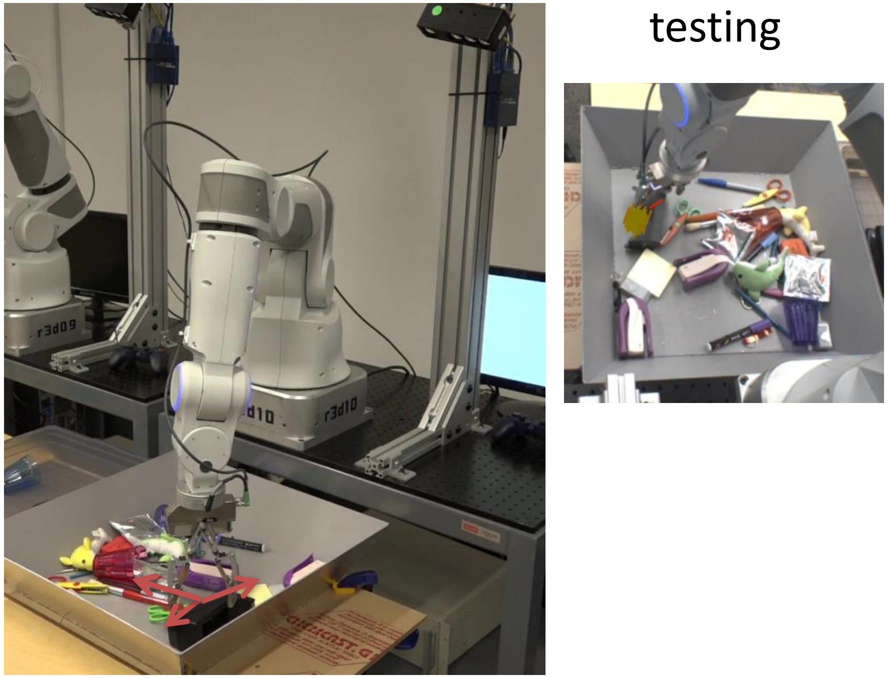
L., Pastor, Krizhevsky, Quillen '16

Using Grasp Success Prediction





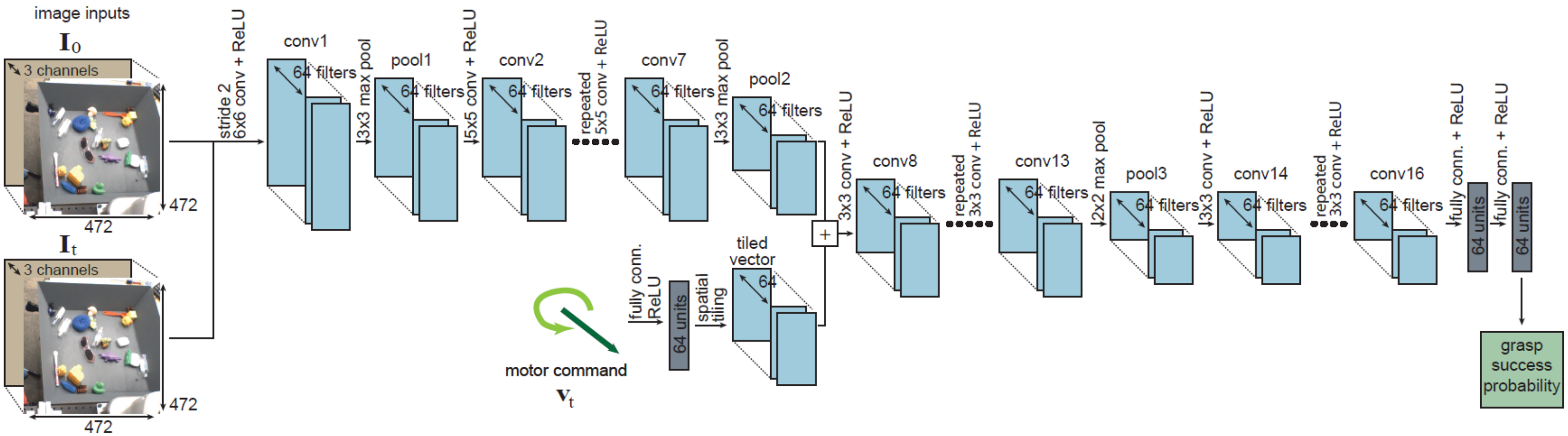
training

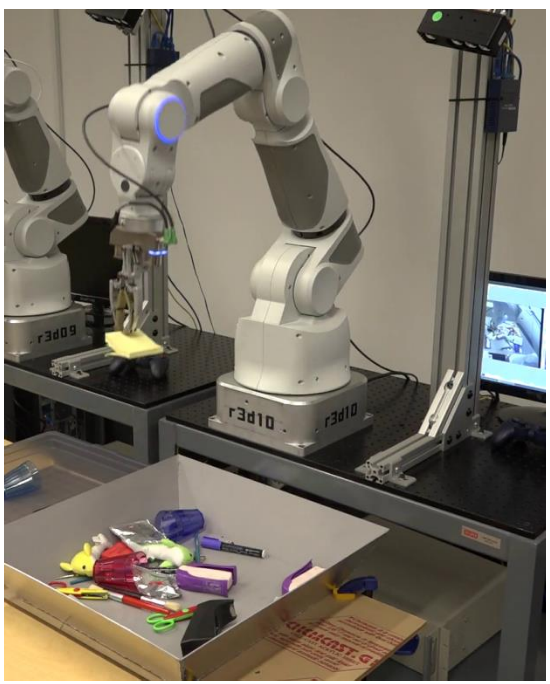


testing

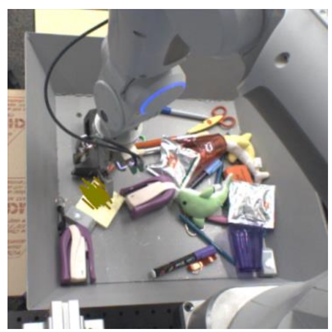
L., Pastor, Krizhevsky, Quillen '16

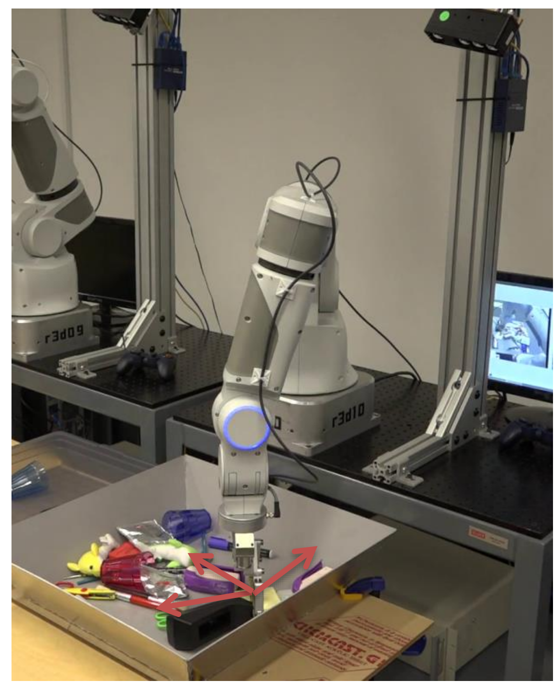
Using Grasp Success Prediction



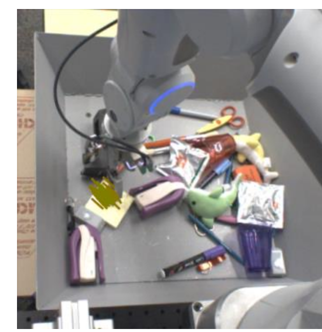


training



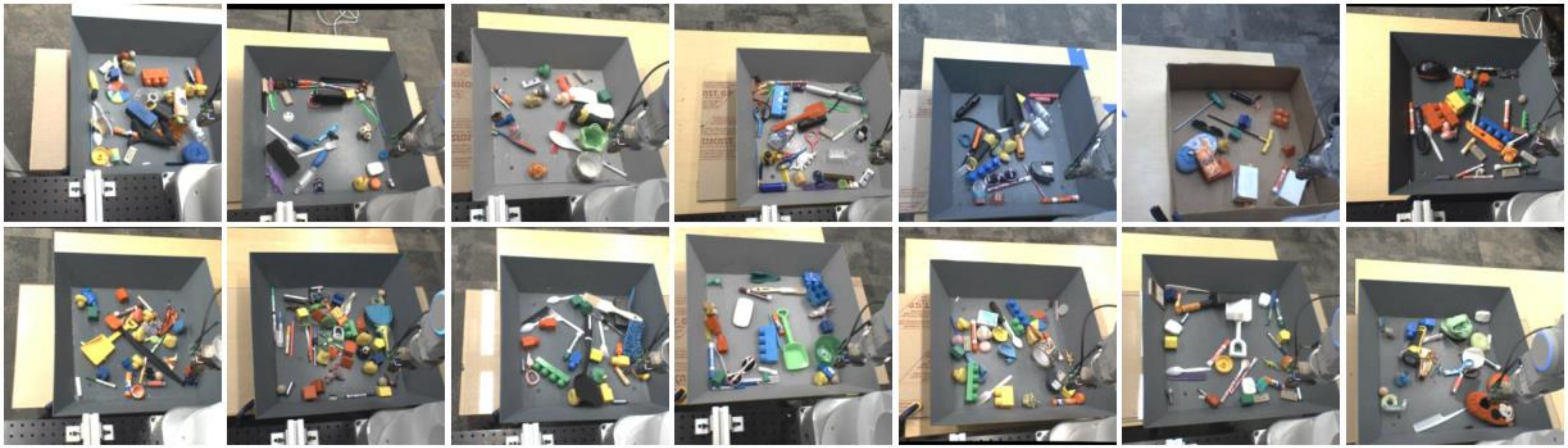


testing

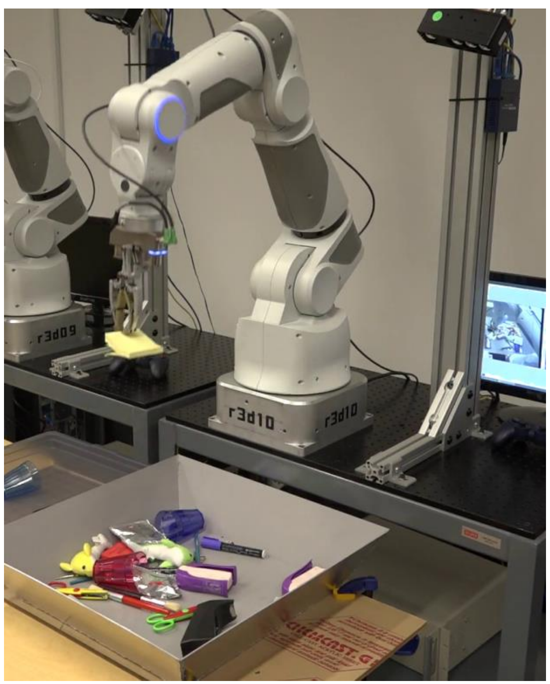


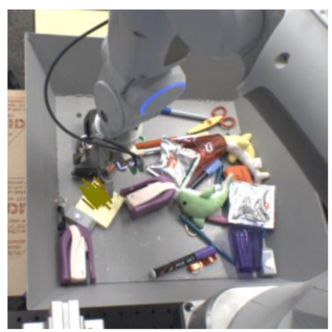
L., Pastor, Krizhevsky, Quillen ‘16

Using Grasp Success Prediction

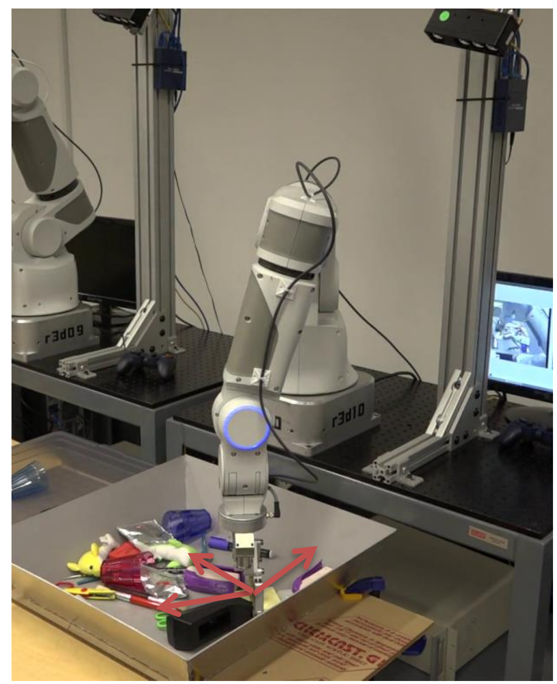


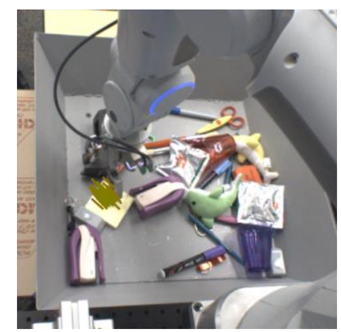
training





testing





L., Pastor, Krizhevsky, Quillen '16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping

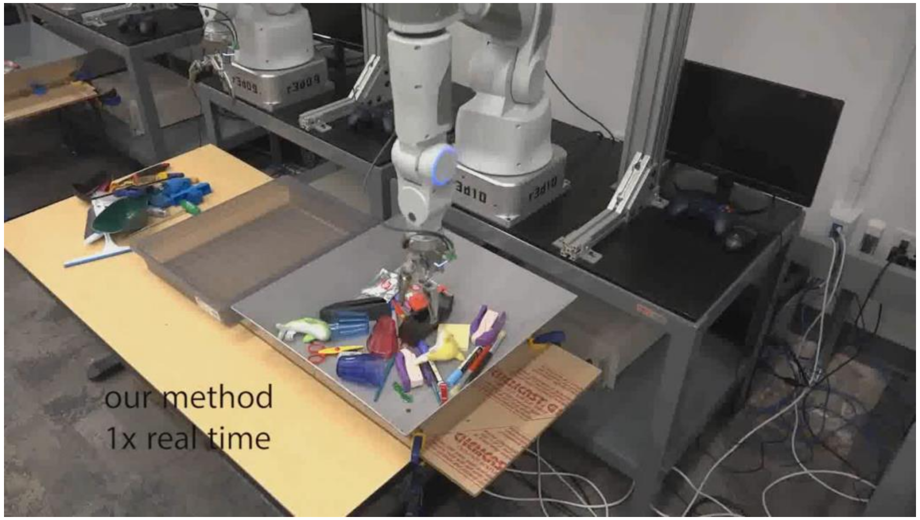
closed-loop grasping

L., Pastor, Krizhevsky, Quillen ‘16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping

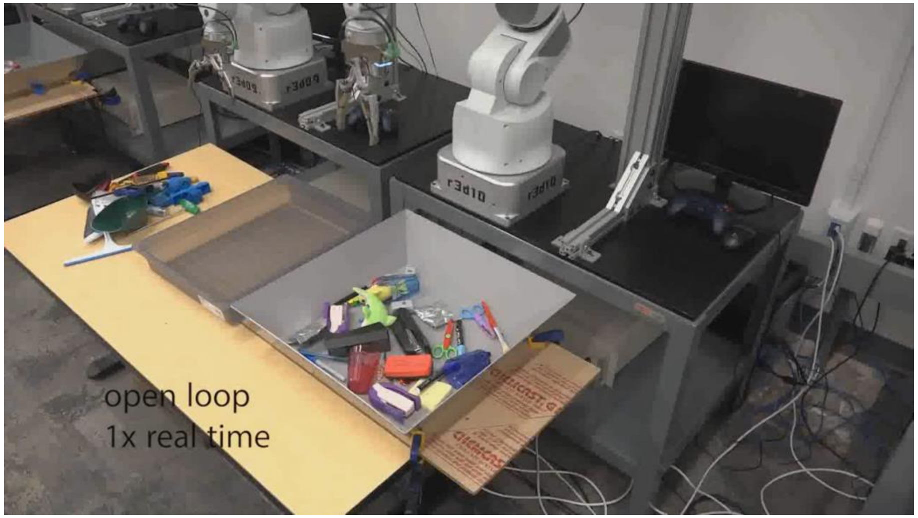
closed-loop grasping



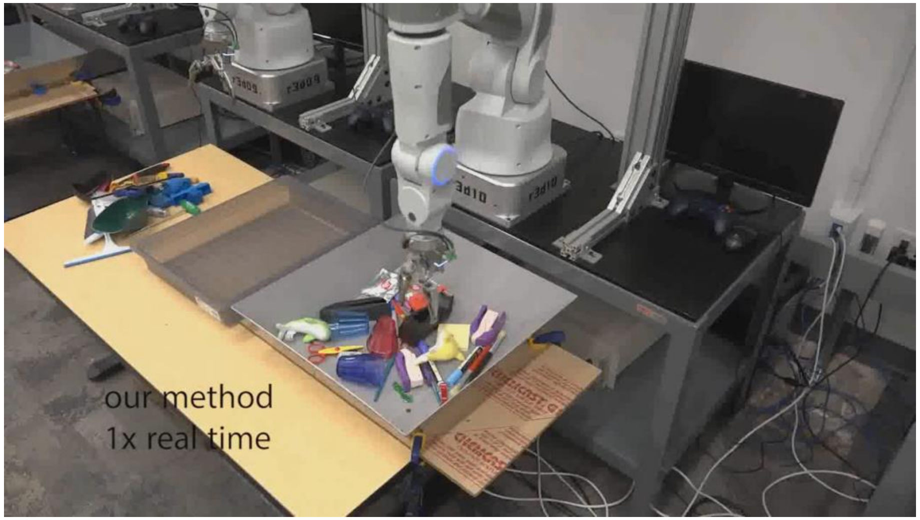
L., Pastor, Krizhevsky, Quillen ‘16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping



closed-loop grasping

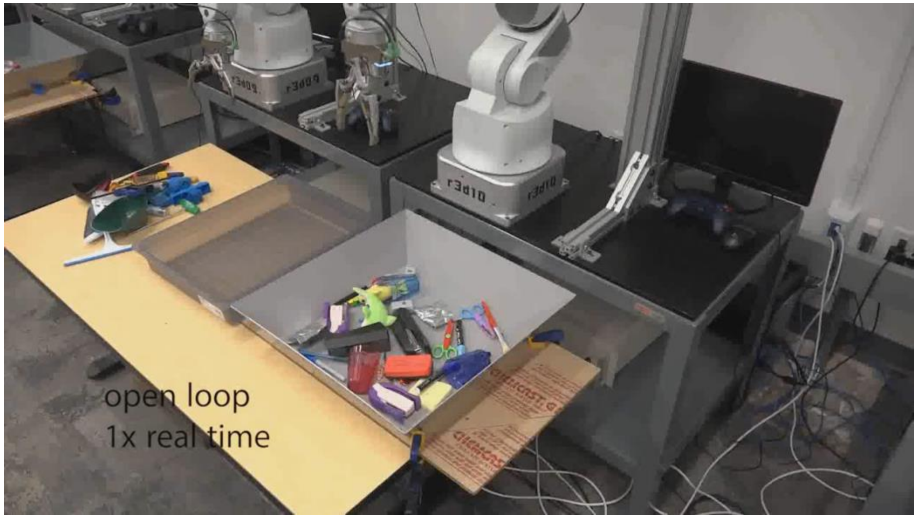


Pinto & Gupta, 2015

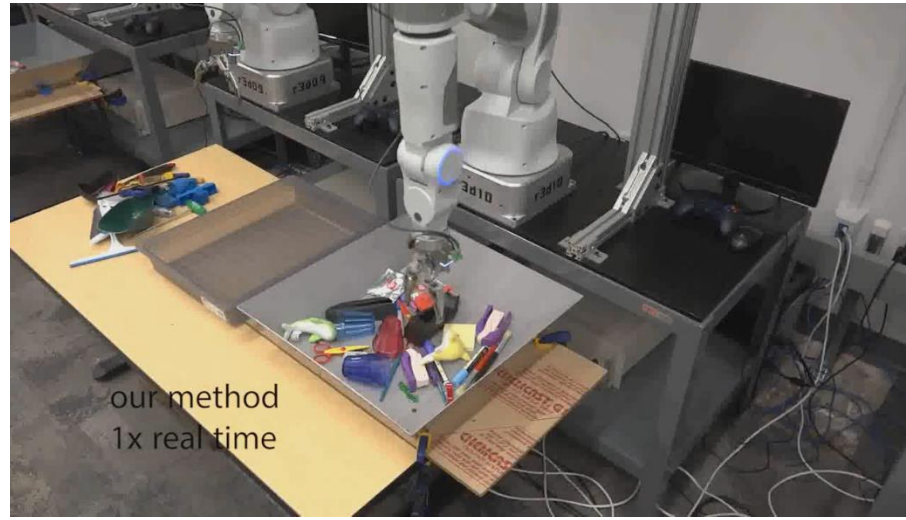
L., Pastor, Krizhevsky, Quillen ‘16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping



closed-loop grasping

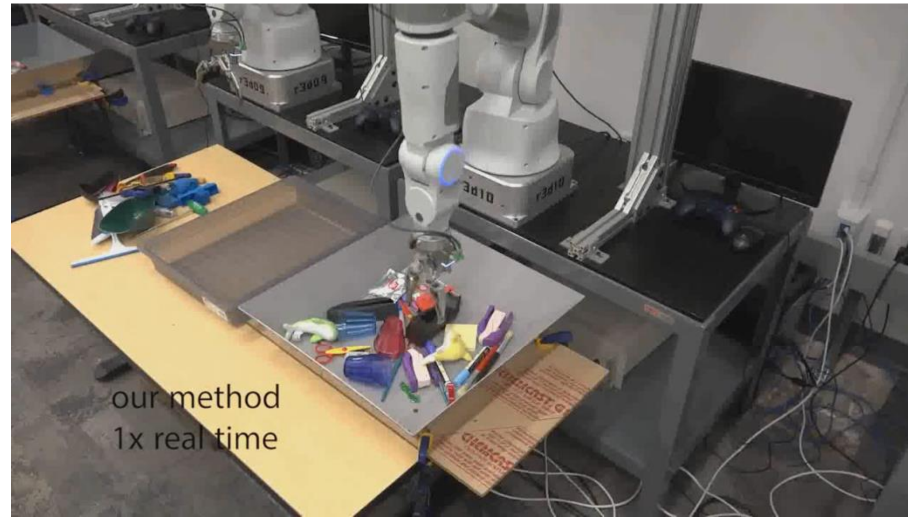


L., Pastor, Krizhevsky, Quillen '16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping

closed-loop grasping

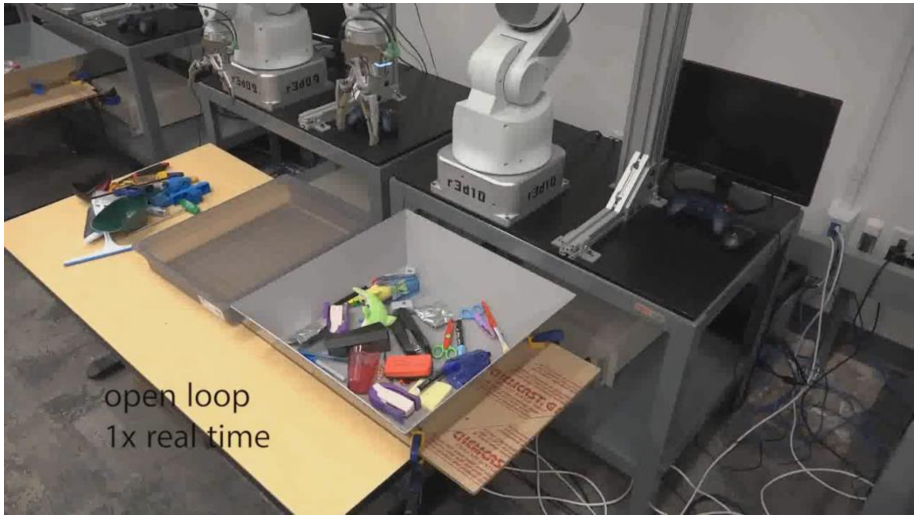


failure rate: 33.7%

L., Pastor, Krizhevsky, Quillen '16

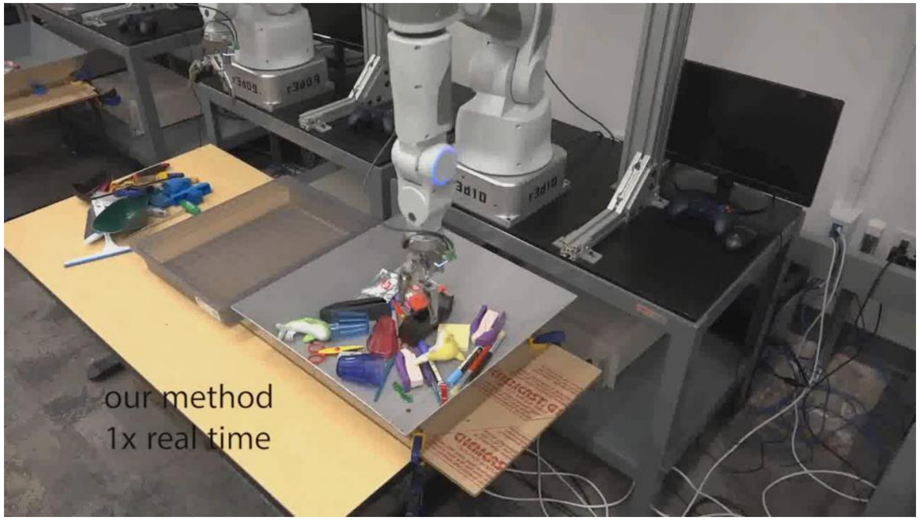
Open-Loop vs. Closed-Loop Grasping

open-loop grasping



failure rate: 33.7%

closed-loop grasping

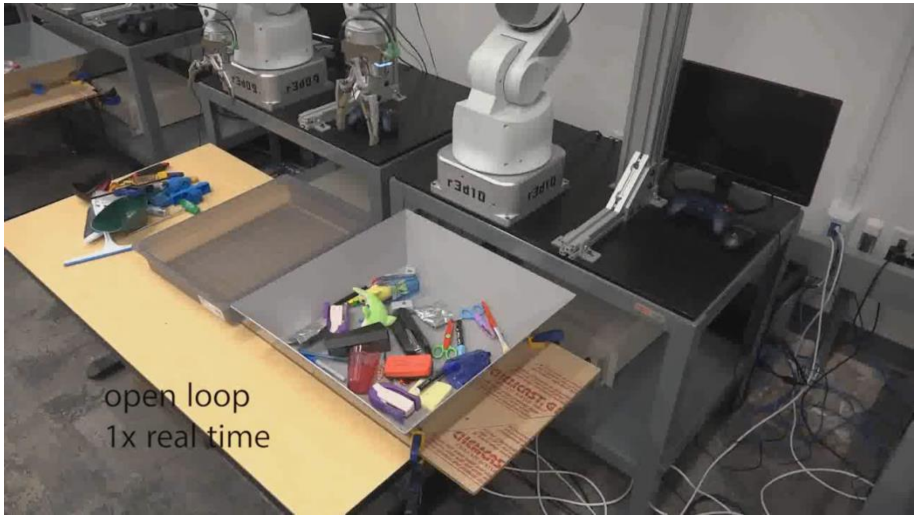


failure rate: 17.5%

L., Pastor, Krizhevsky, Quillen '16

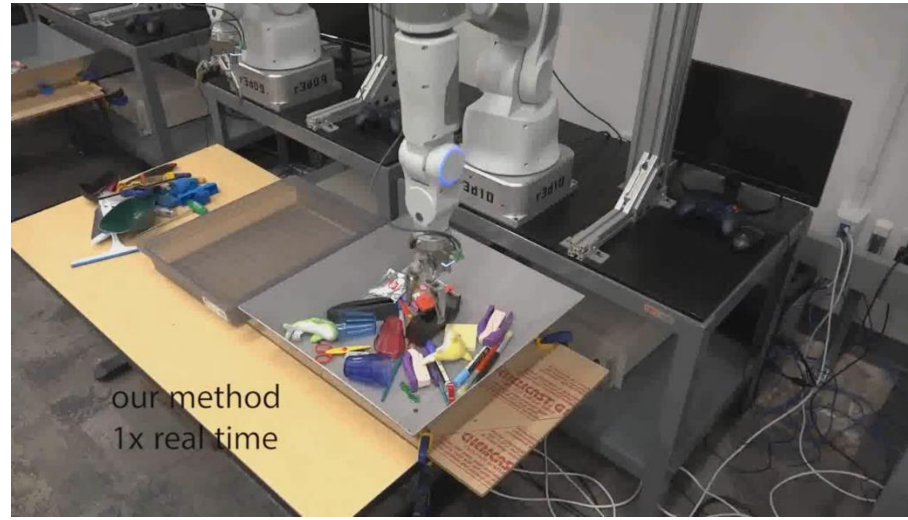
Open-Loop vs. Closed-Loop Grasping

open-loop grasping



failure rate: 33.7%

closed-loop grasping



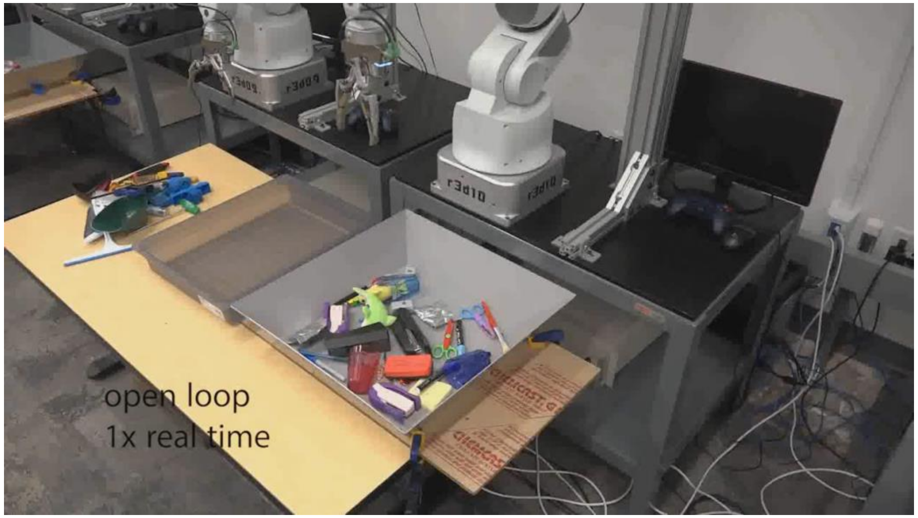
failure rate: 17.5%

depth + segmentation
failure rate: 35%

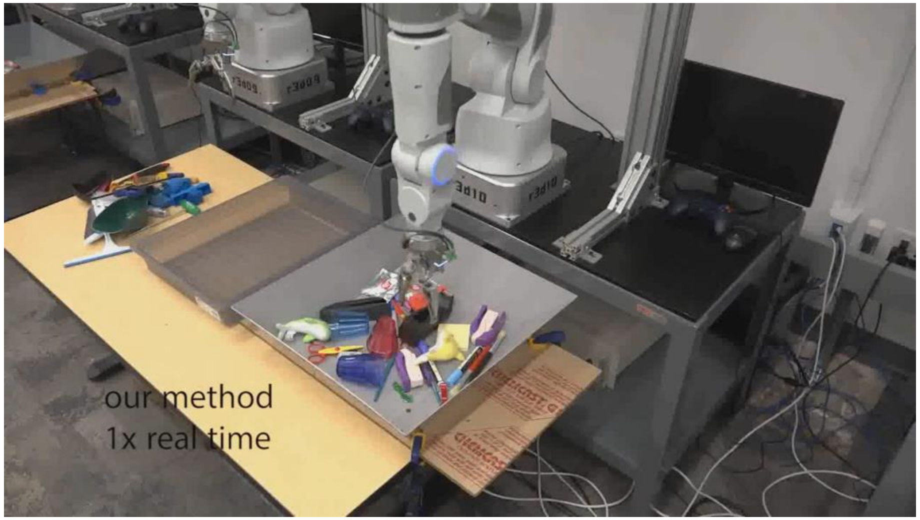
L., Pastor, Krizhevsky, Quillen '16

Open-Loop vs. Closed-Loop Grasping

open-loop grasping



closed-loop grasping



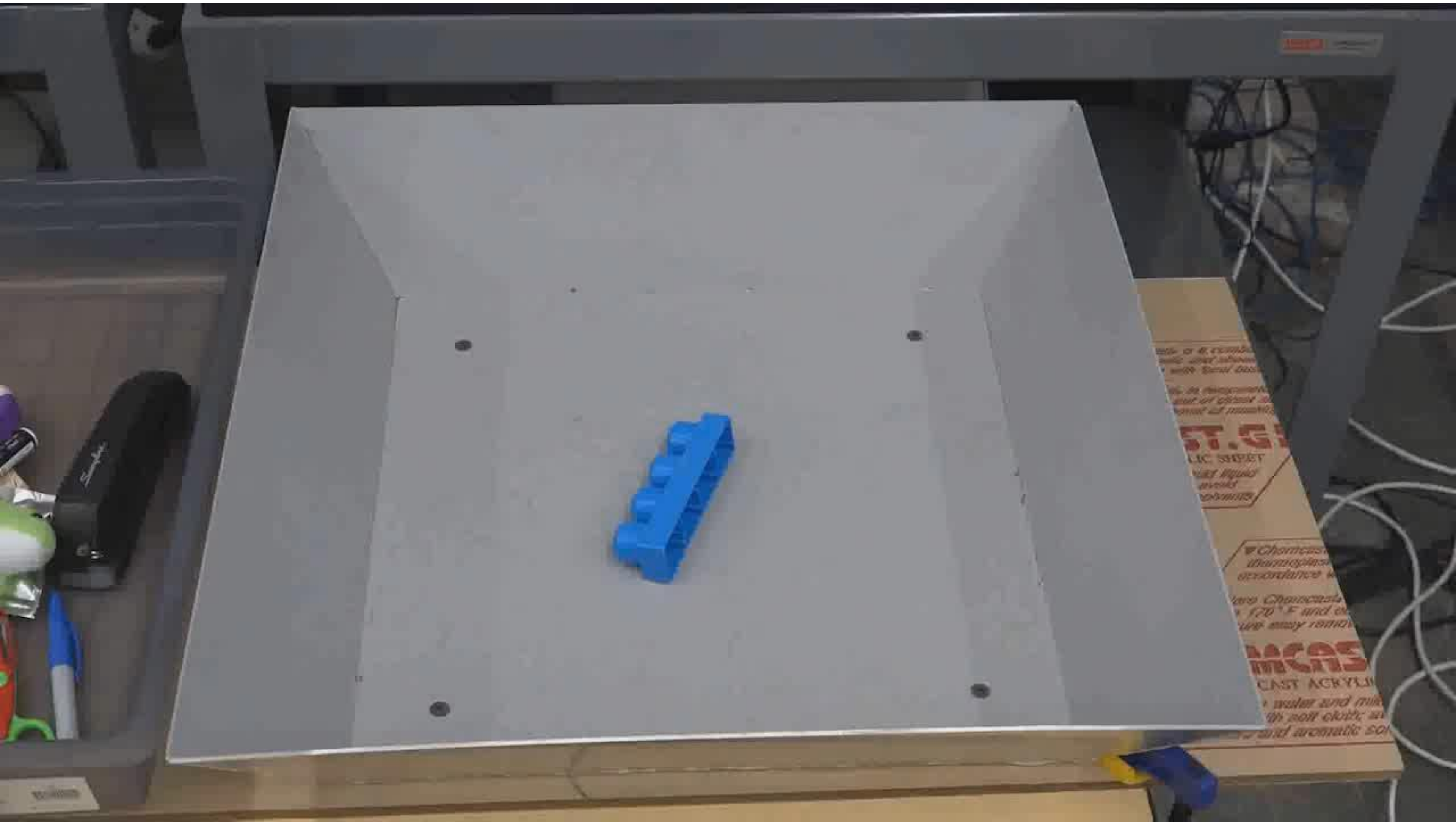
failure rate: 33.7%

depth + segmentation
failure rate: 35%

failure rate: 17.5%

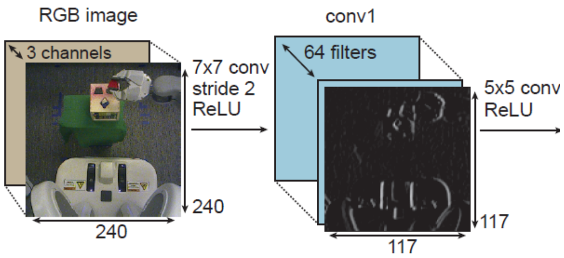
L., Pastor, Krizhevsky, Quillen '16

Grasping Experiments



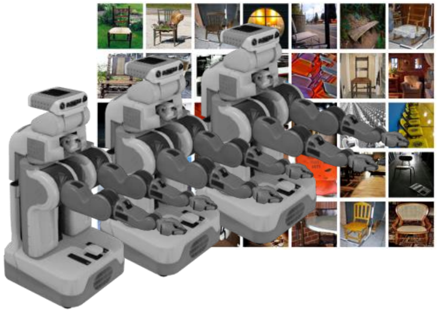
L., Pastor, Krizhevsky, Quillen '16

Overview



Training visuomotor policies

Deep robotic learning at scale



Future directions

ingredients for success in learning:

supervised learning:	learning sensorimotor skills:
✓ computation	✓ computation
✓ algorithms	~ algorithms
✓ data	? data

ingredients for success in learning:

supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

learning sensorimotor skills:

- ✓ computation
- ~ algorithms
- ? data

ingredients for success in learning:

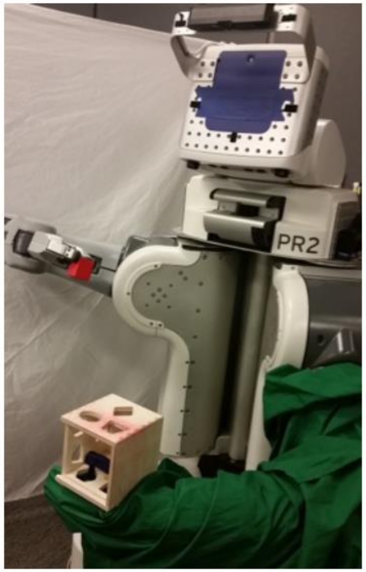
supervised learning:

- ✓ computation
- ✓ algorithms
- ✓ data

learning sensorimotor skills:

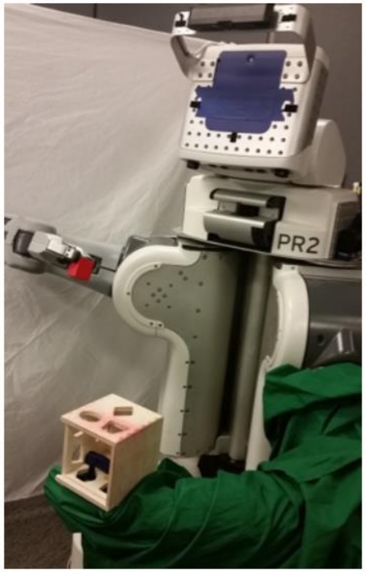
- ✓ computation
- ~ algorithms
- ? data

Learning what Success Means



Finn, L., Abbeel '16

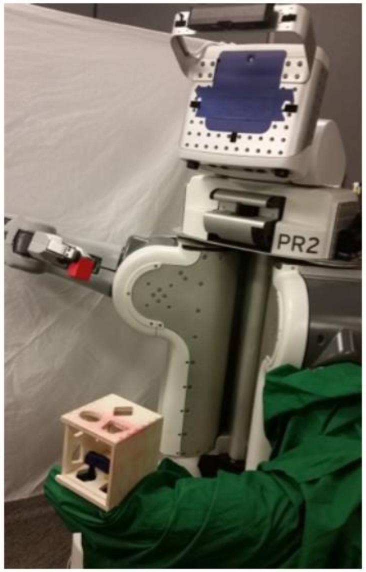
Learning what Success Means

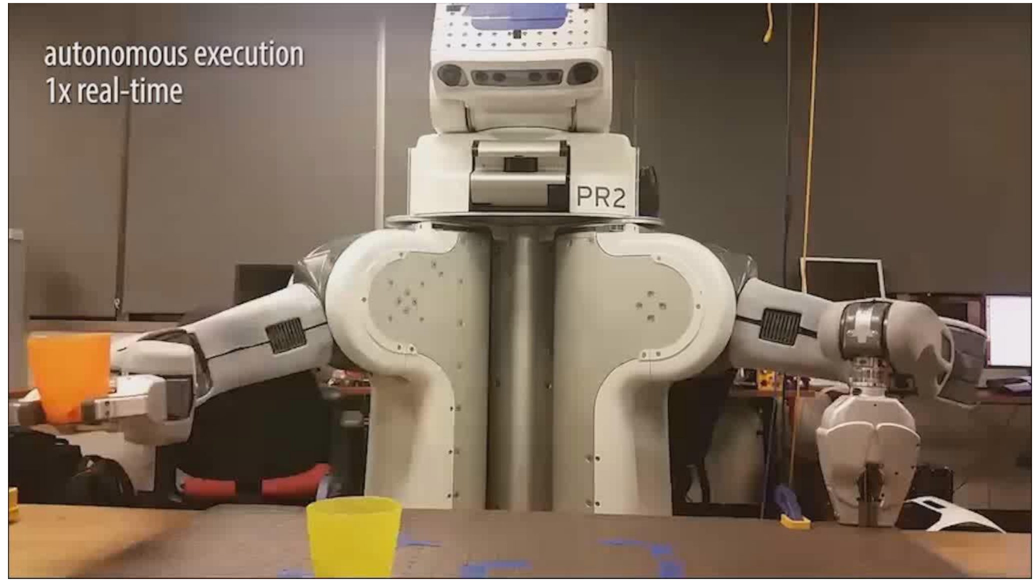


$$c(\mathbf{x}, \mathbf{u}) =$$
$$w_1 f_{\text{target}}(\mathbf{x}) +$$
$$w_2 f_{\text{torque}}(\mathbf{u})$$

Finn, L., Abbeel '16

Learning what Success Means

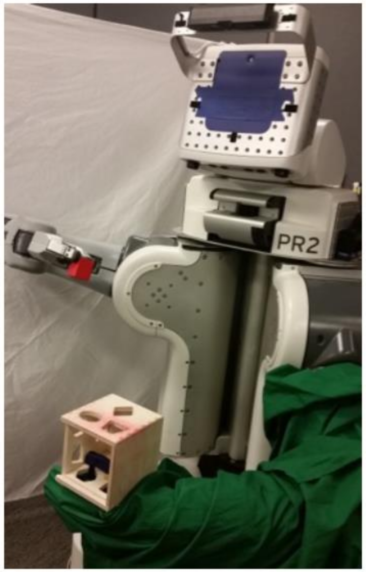




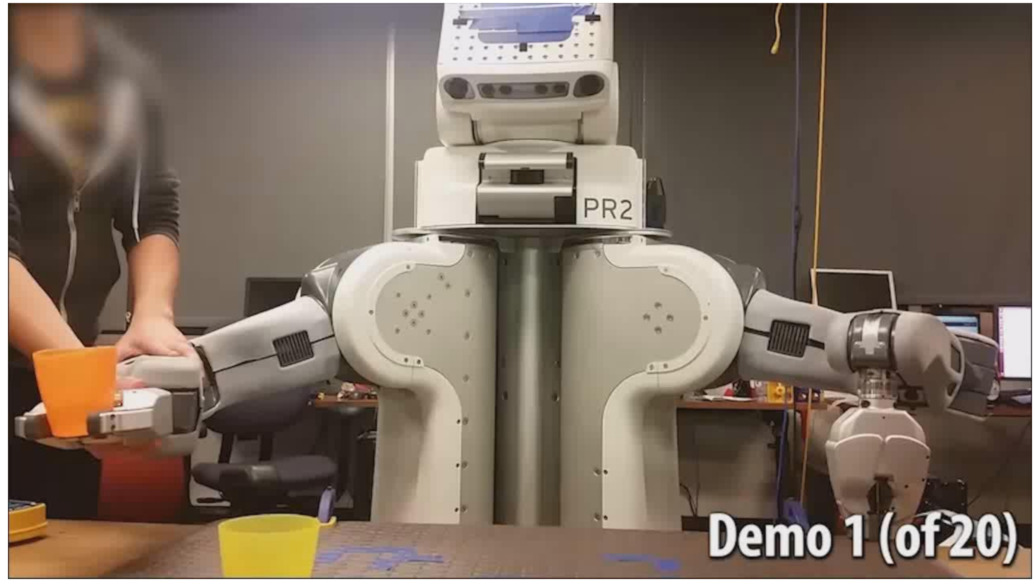
$$c(\mathbf{x}, \mathbf{u}) =$$
$$w_1 f_{\text{target}}(\mathbf{x}) +$$
$$w_2 f_{\text{torque}}(\mathbf{u})$$

Finn, L., Abbeel '16

Learning what Success Means



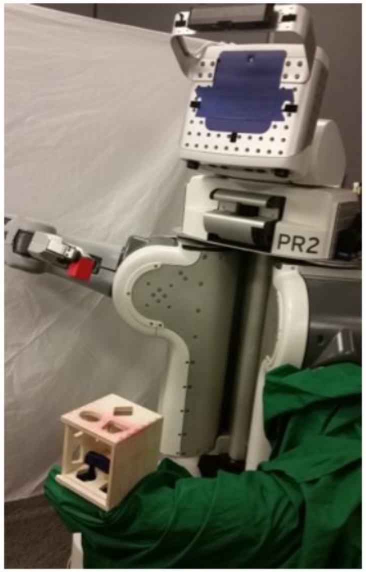
$$c(\mathbf{x}, \mathbf{u}) =$$
$$w_1 f_{\text{target}}(\mathbf{x}) +$$
$$w_2 f_{\text{torque}}(\mathbf{u})$$



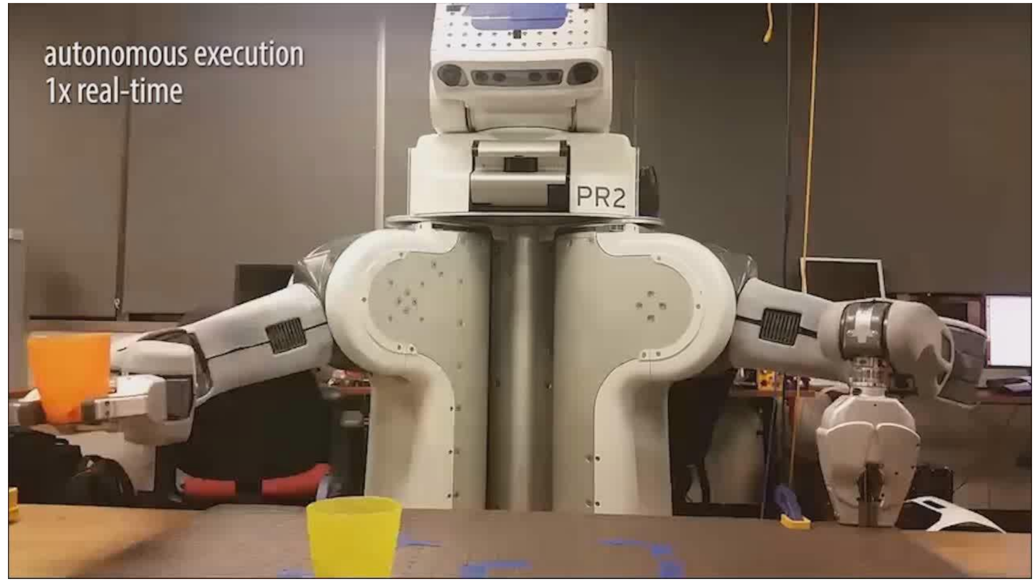
can we *learn* the cost with visual features?

Finn, L., Abbeel '16

Learning what Success Means



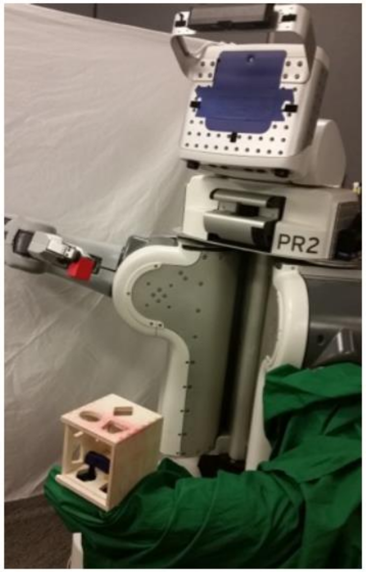
$$c(\mathbf{x}, \mathbf{u}) =$$
$$w_1 f_{\text{target}}(\mathbf{x}) +$$
$$w_2 f_{\text{torque}}(\mathbf{u})$$

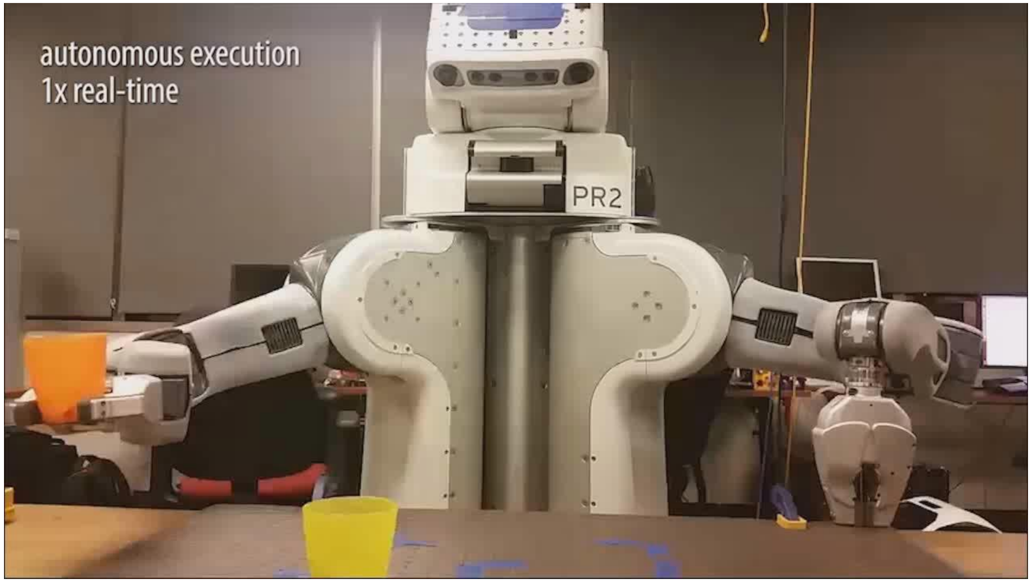


can we *learn* the cost with visual features?

Finn, L., Abbeel '16

Learning what Success Means



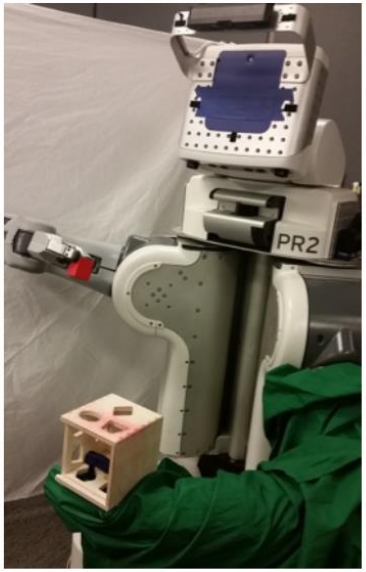


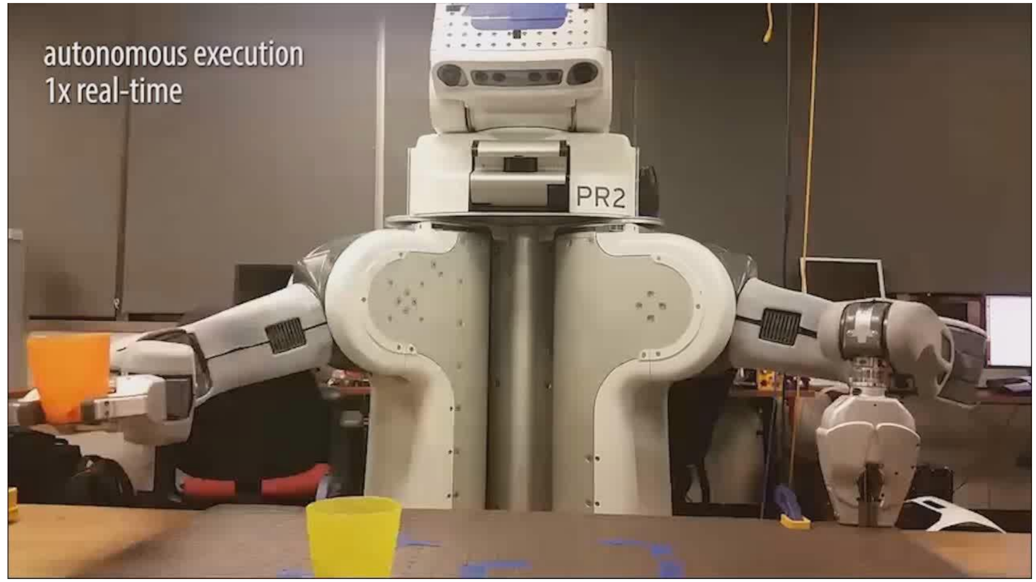
$$c(\mathbf{x}, \mathbf{u}) =$$
$$w_1 f_{\text{target}}(\mathbf{x}) +$$
$$w_2 f_{\text{torque}}(\mathbf{u})$$

can we *learn* the cost with visual features?

Finn, L., Abbeel '16

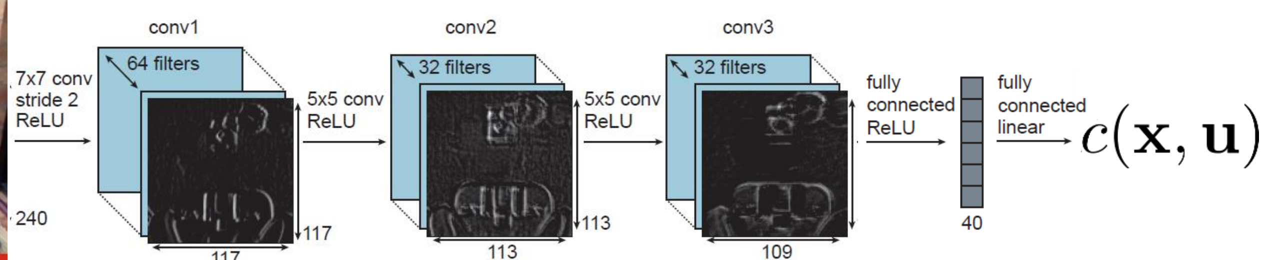
Learning what Success Means





$$c(\mathbf{x}, \mathbf{u}) = w_1 f_{\text{target}}(\mathbf{x}) + w_2 f_{\text{torque}}(\mathbf{u})$$

can we *learn* the cost with visual features?

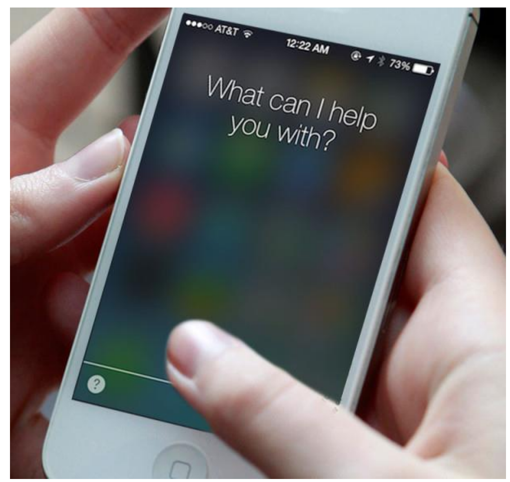


Finn, L., Abbeel '16

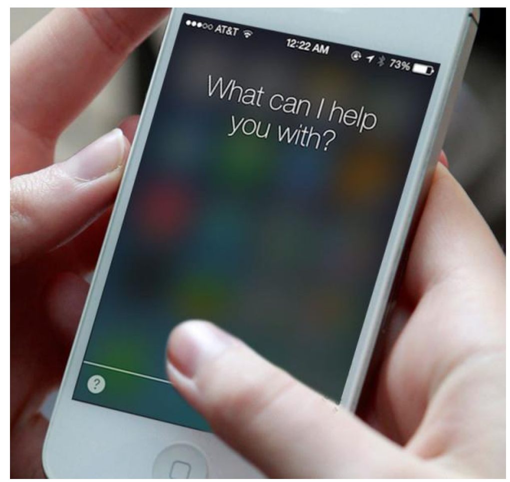
Broader implications: end-to-end learning for decision making in the real world

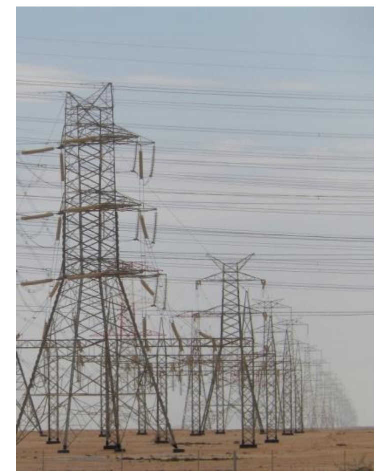
Broader implications: end-to-end learning for decision making in the real world

Broader implications: end-to-end learning for decision making in the real world



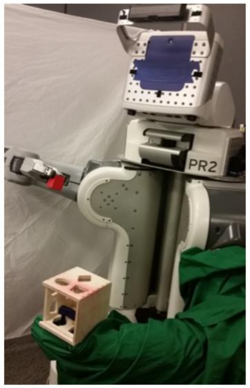
Broader implications: end-to-end learning for decision making in the real world





Acknowledgements

BRETT

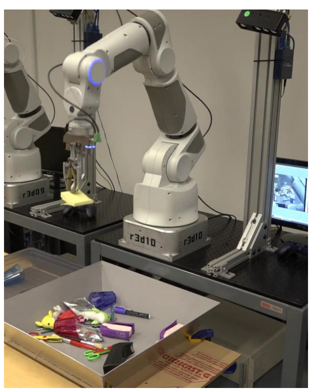


Chelsea Finn

Trevor Darrell

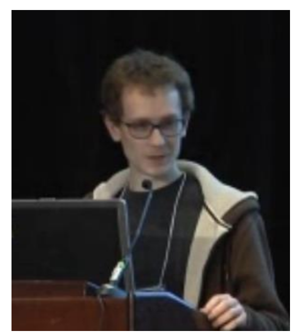
Pieter Abbeel

r3d10



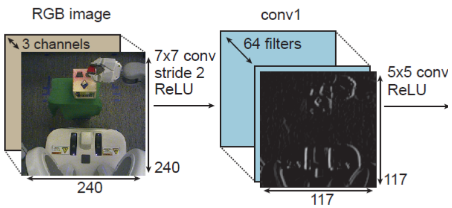
Peter Pastor

Alex Krizhevsky



Deirdre Quillen

Questions?



Bibliography:

Levine, Pastor, Krizhevsky. "Learning Hand-Eye Cordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection." Technical report. 2016.
Finn, Levine, Abbeel. "Guided Cost Learning: Inverse Optimal Control via Policy Optimization." Under review. 2016.
Zhang, Kahn, Levine, Abbeel. "Learning Deep Control Policies for Autonomous Aerial Vehicles with MPC-Guided Policy Search." ICRA. 2016.
Kumar, Todorov, Levine. "Optimal Control with Learned Local Models: Application to Dexterous Manipulation." ICRA.2016.
Levine*, Finn*, Darrell, Abbeel. "End-to-End Training of Deep Visuomotor Policies." arXiv 1504.00702. 2015.
Levine, Wagener, Abbeel. "Learning Contact-Rich Manipulation Skills with Guided Policy Search." ICRA. 2015.
Levine, Abbeel. "Learning Neural Network Policies with Guided Policy Search under Unknown Dynamics." Neural Information Processing Systems (NIPS). 2014.

website: http://homes.cs.washington.edu/~svlevine/