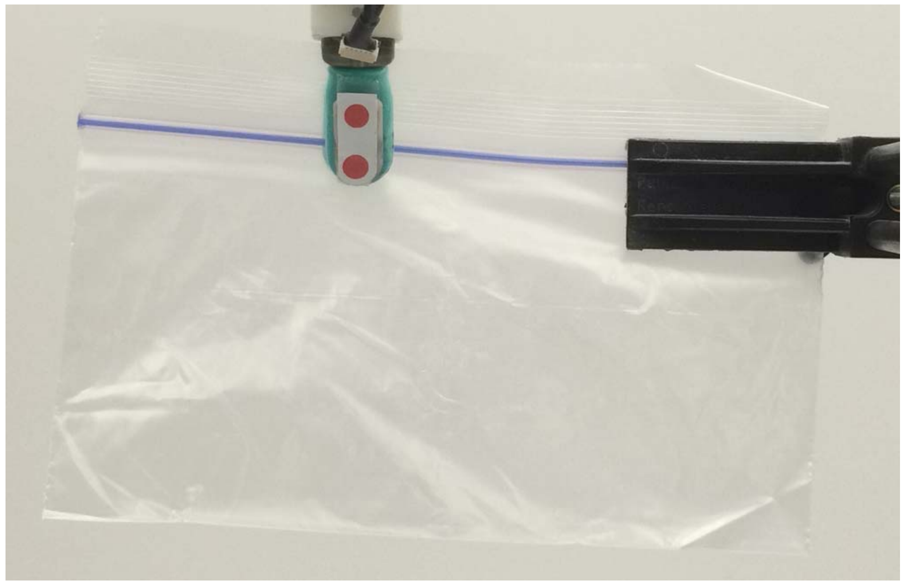
From tactile sensor data to haptic percepts and task-driven decisions



Veronica J. Santos

Assoc. Prof. of Mechanical and Aerospace Engineering

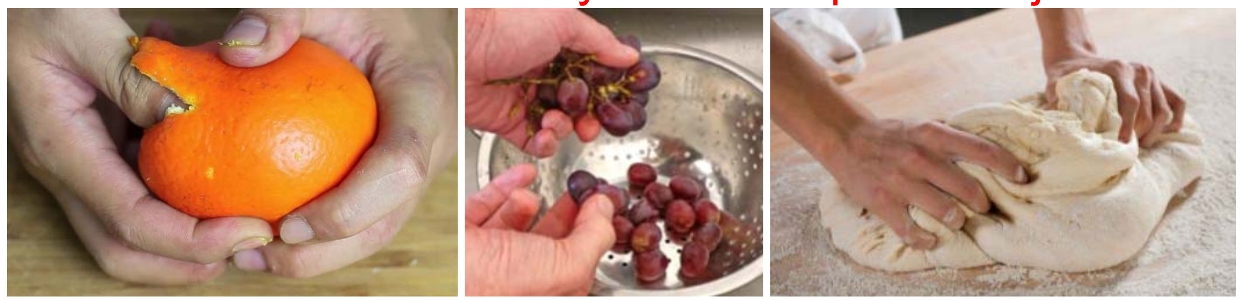
BiomechatronicsLab.ucla.edu

November 15, 2016

No line of sight

Occluded by a hand or part of object

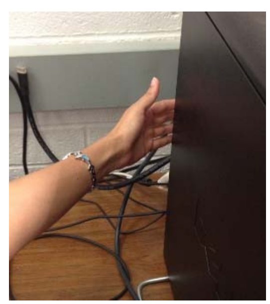
In the dark



Deformable…

…and animate

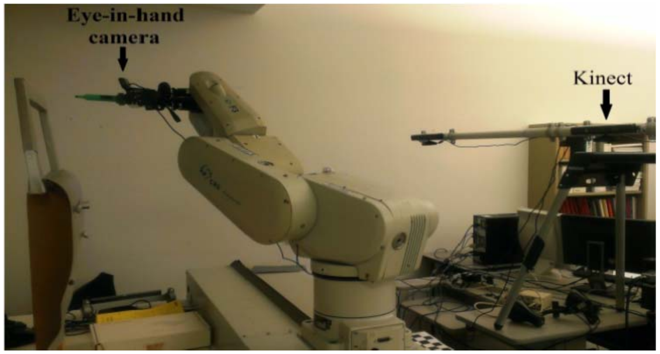
Inside containers



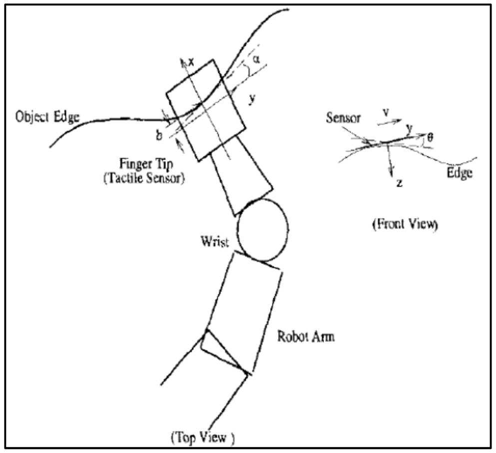
Around obstacles

Extreme!

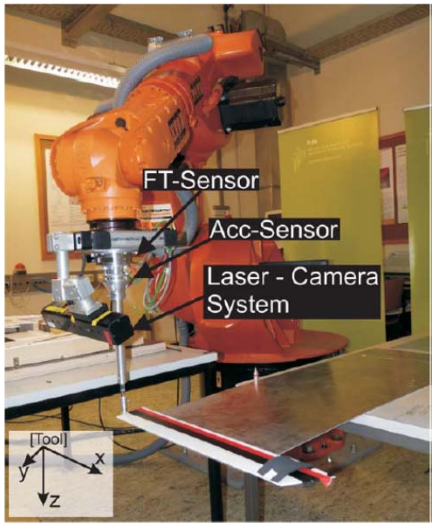
Contour-following and edge-tracking



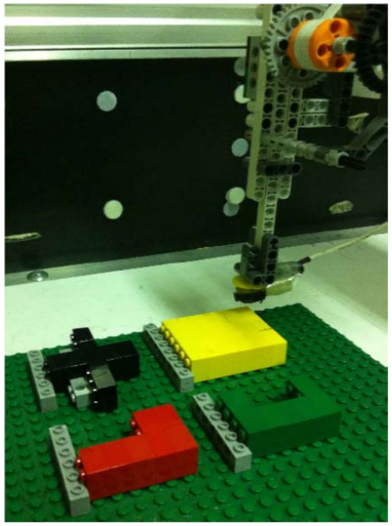
Visual servoing
(e.g. Nakhaeinia et al., 2014)



Tactile servoing
(e.g. Chen et al., 1995)



Vision, force, and accel.
(Koch et al., 2013)



Probabilistic active tactile perception
(Martinez-Hernandez et al., 2013)

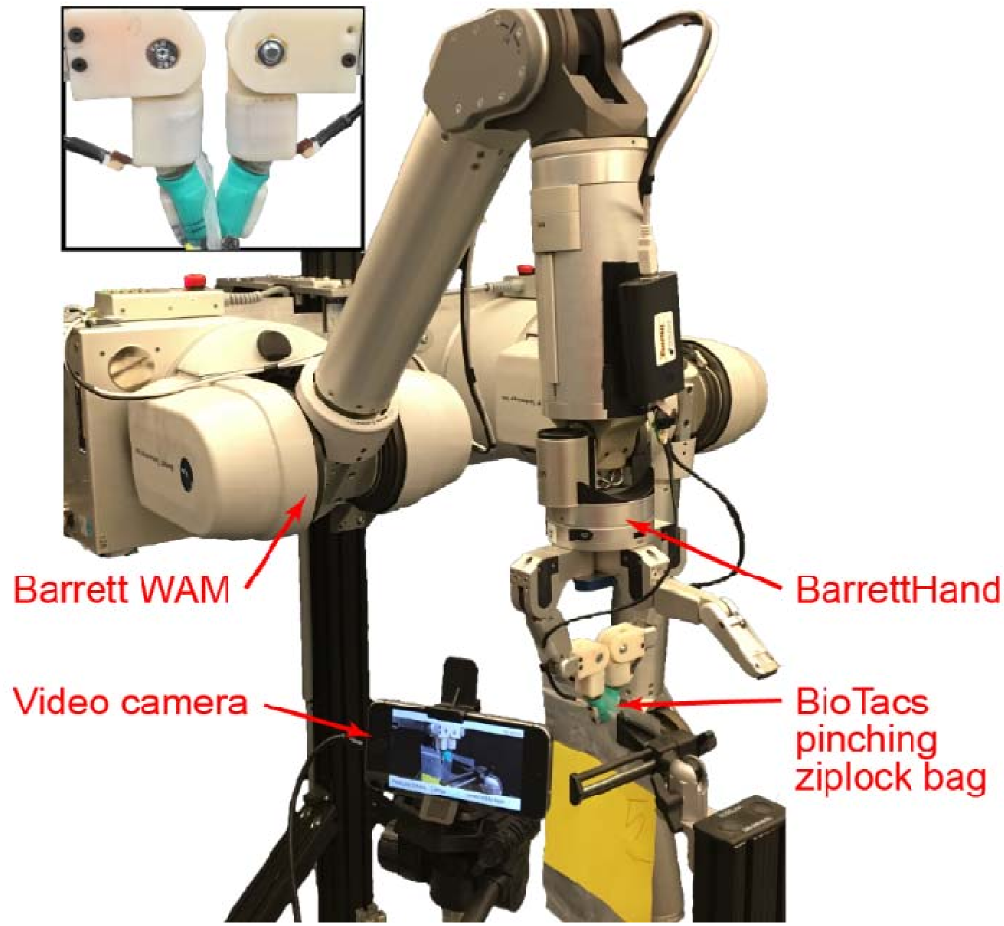
A twist on the traditional contour-following task

Goal:

Learn to close a ziplock bag using touch and proprioception alone.

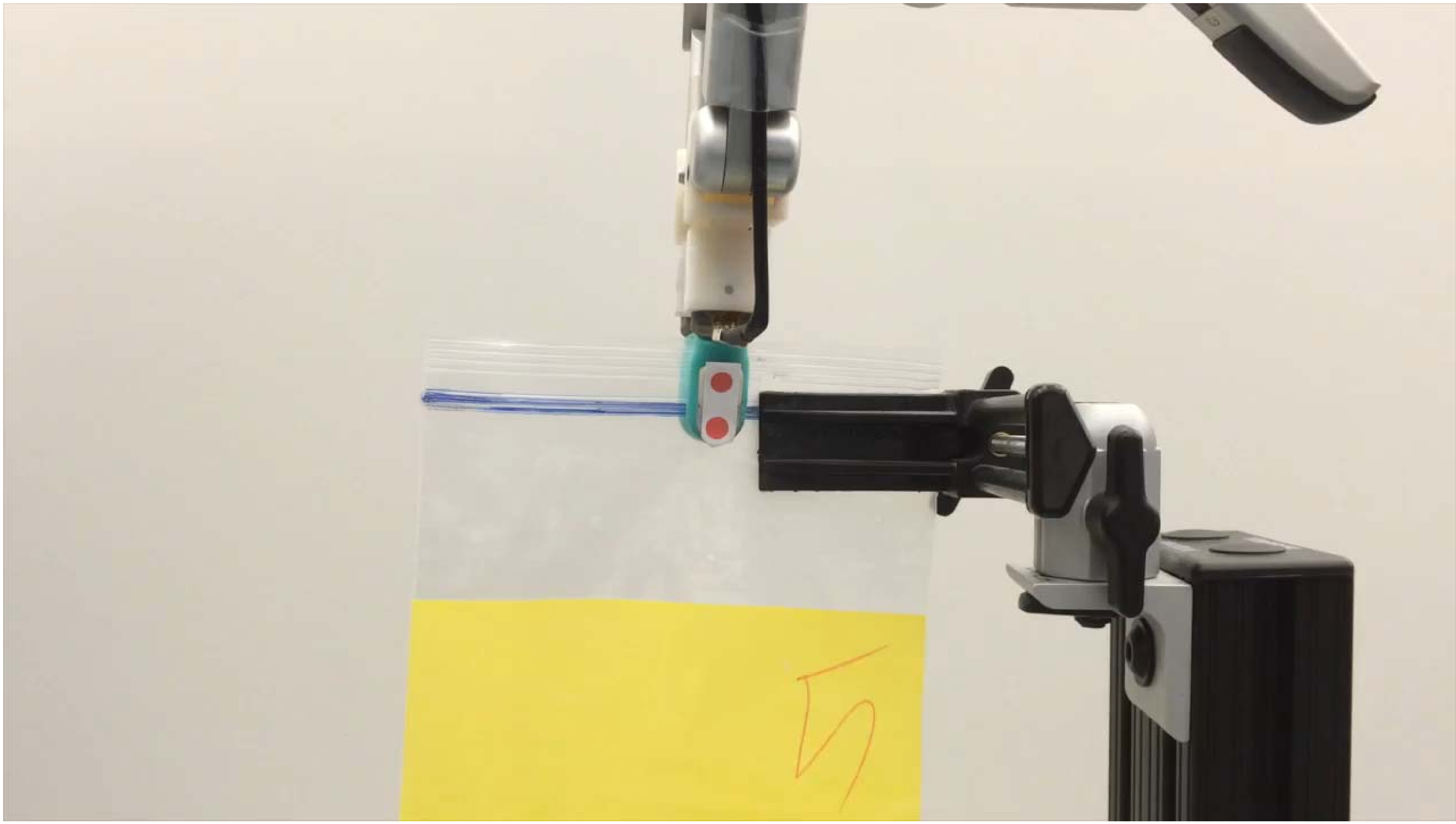
Challenge:

Manipulation of a transparent, deformable object whose functional features are occluded by the fingertips.



Hellman, R.B., ‘Haptic Perception, Decision-making, and Learning for Manipulation with Artificial Hands’, Arizona State University, Tempe, AZ, Aug. 2016.

Testing with preplanned trajectories and without closed-loop haptic perception



Reinforcement learning

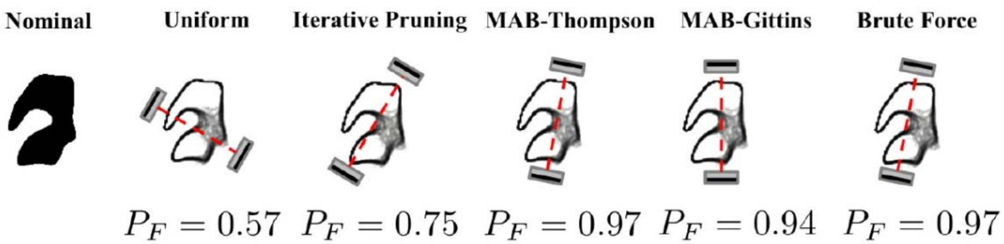
- *Exploration* (trial and error) is used to learn how different actions are rewarded from a given state.
- *Exploitation* is used to select actions based on a policy and typically only occurs once the state-action space has been reasonably mapped out (i.e. learned).
- We considered two reinforcement learning algorithms
  - Q-learning (benchmark)
  - Contextual Multi-armed Bandits
    - Variant: Single agent learner with uniform partitions

Multi-Armed Bandits (MABs)

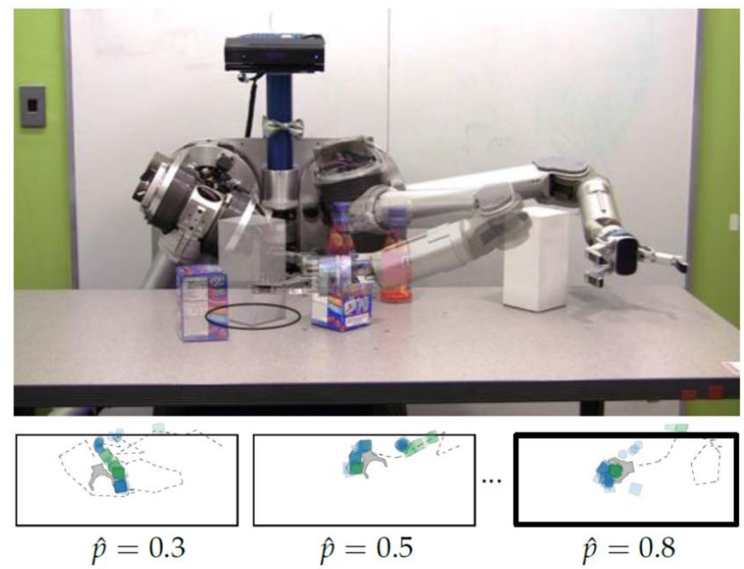
- Given limited resources (hardware life, researcher time), what actions should you take?
  - Tactile data are expensive!

- Benefits of MABs:
  - Can balance exploration vs. exploitation of the state-action space during policy learning.
  - Guaranteed to minimize the total regret given a finite time horizon.

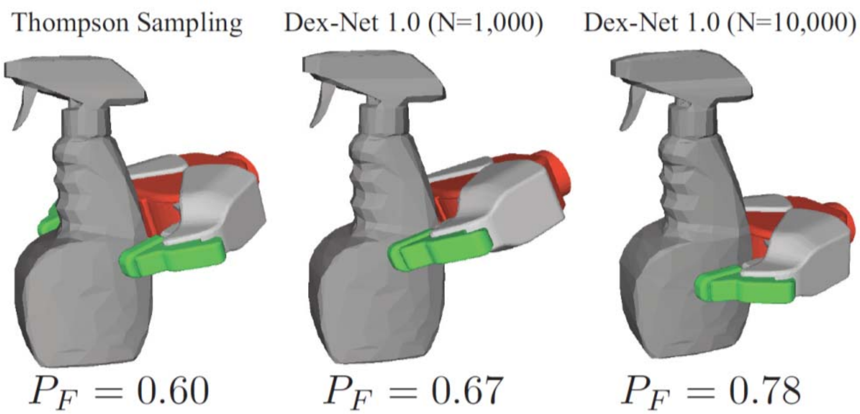
Multi-Armed Bandit models for robot planning



2D grasp planning w/ uncertainty
(Laskey et al., 2015)



Trajectory selection for rearrangement planning w/ uncertainty
(Koval et al., 2015)



3D grasp planning w/ uncertainty
(Mahler et al., 2016)

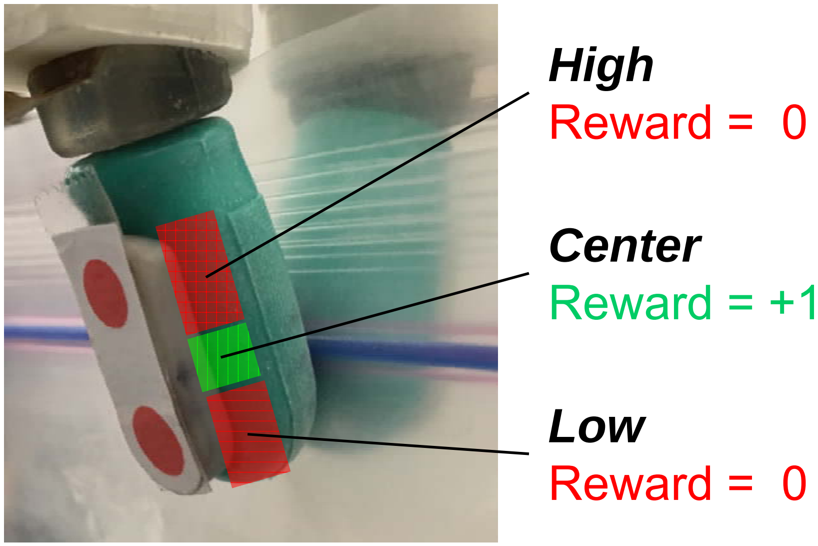
Contextual Multi-Armed Bandits (C-MABs)

- Contextual MABs allow for multiple states or "contexts," each of which has its own set of action-reward relationships.
- *Exploration:* Each context has its own action counters that track how many times an action has been tried.
- *Exploitation:* Can occur during training if all actions for a given context have been explored sufficiently.
- C-MABs balance exploration with exploitation in order to minimize cumulative regret.
  – Exploration vs. exploitation is decided by a control function $D(t)$ that is a function of the current time t, the similarity within the state space, and the dimensionality of the action space.

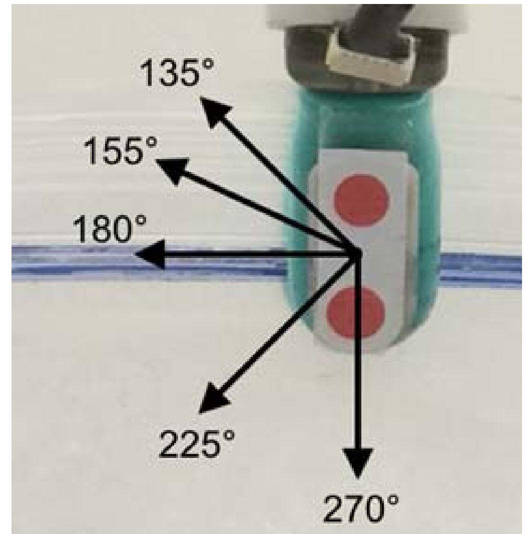

Collaboration with **C. Tekin and M. Van der Schaar**, authors of "Distributed Online Learning via Cooperative Contextual Bandits." *IEEE Trans Signal Proc*, 2015.

Preparations for reinforcement learning

States and Rewards



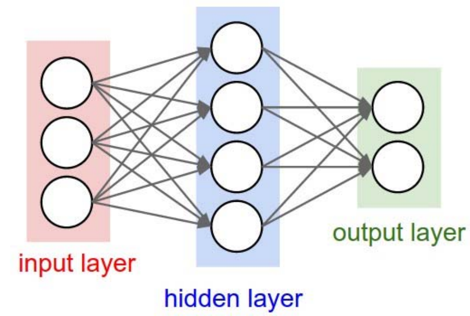
Actions



- Since the zipper contour deforms as the bag is manipulated, we moved the fingertips relative to the zipper.
- Actions were 0.75 cm fingertip movements from the current fingertip location at 0.5 cm/s.
- Fingertip orientation was constant and movements were constrained to the plane of the bag.

States were classified using deep neural nets (DNNs)

- A DNN classifier was trained to fit the nonlinear tactile data using TensorFlow*.
  - Inputs: 19x1 feature vector of normalized changes in impedance electrode data (fingerpad deformation)
  - Outputs: *Low*, *Center*, *High* labels
  - DNN had three hidden layers and 512 nodes per hidden layer.
  - Trained on 7,200 trials (90% of data) and validated with 800 trials (10% of data).

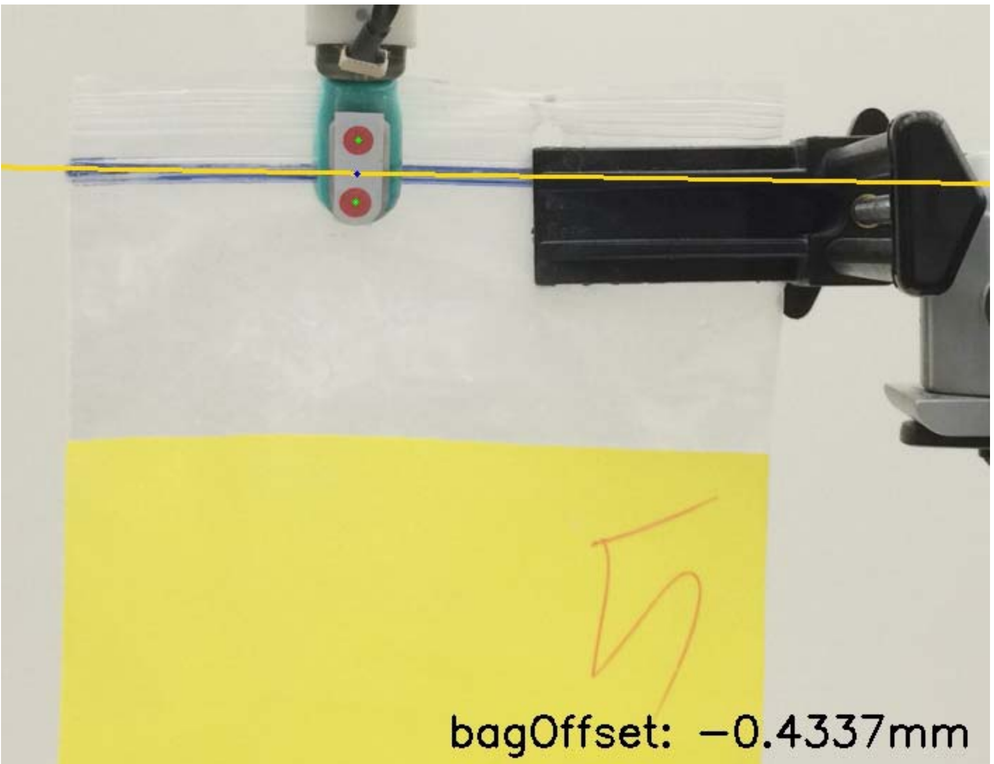


(http://cs231n.github.io)

- The DNN performed with 89% and 86% accuracy on the training and validation datasets, respectively.

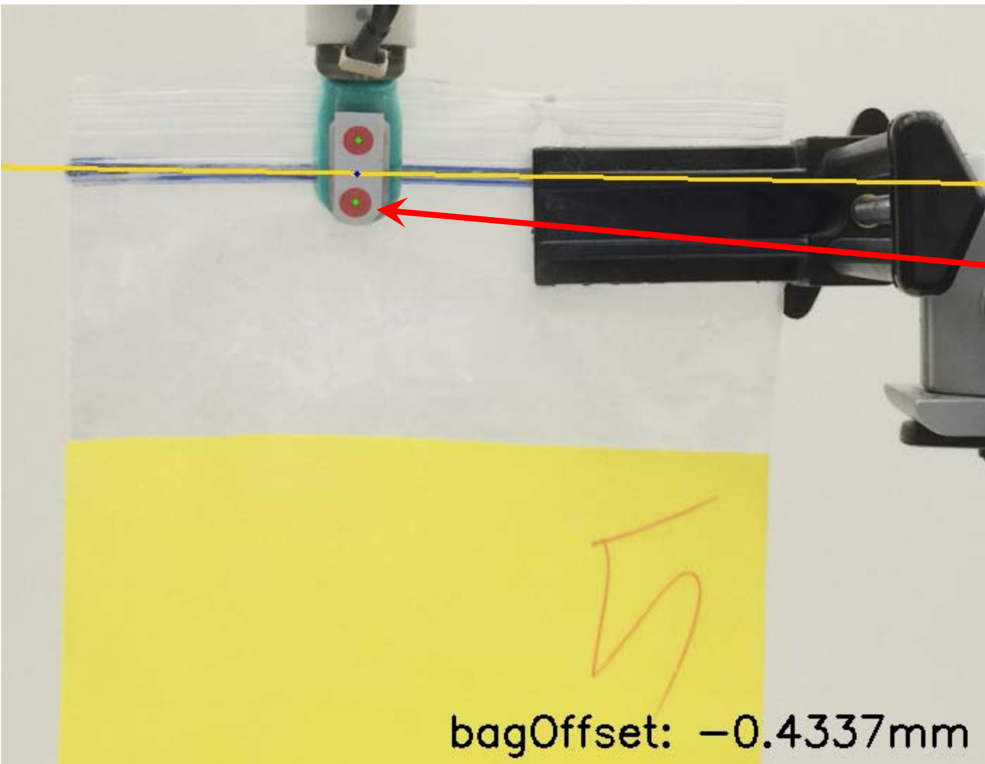
* Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., … Zheng, X. (2016). "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems." arXiv:1603.04467.

Computer vision was used to automatically assign rewards during supervised learning



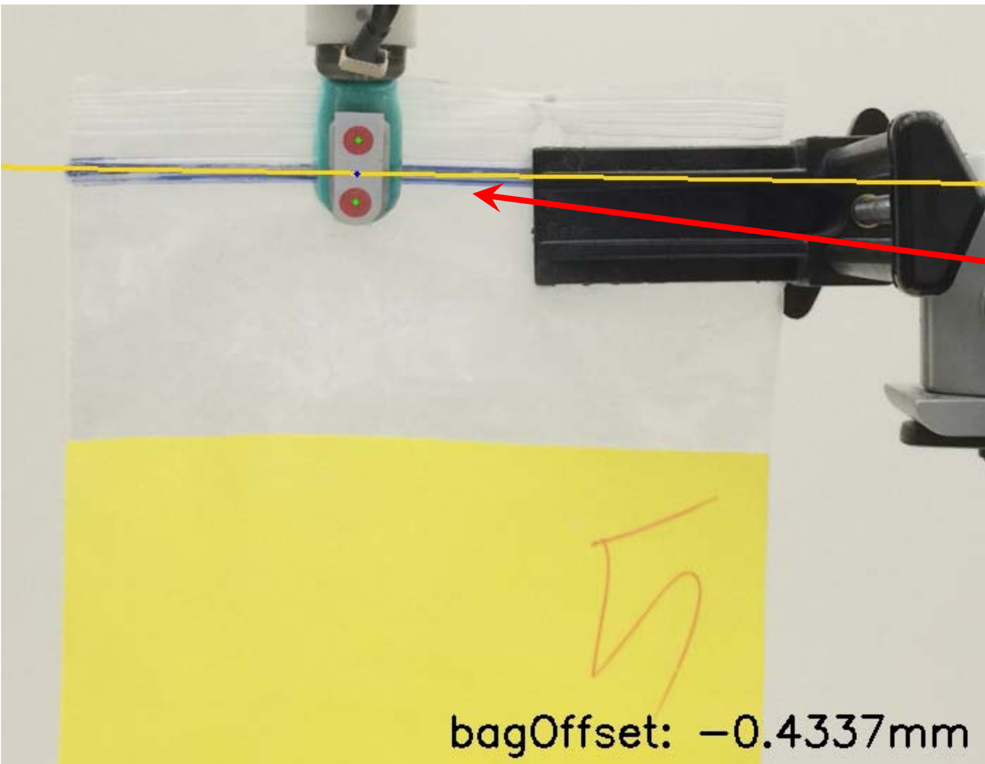
OpenCV was used to autonomously extract the *zipper offset*, the distance between the center of the fingerpad and the estimated location of the zipper along the fingerpad.

Computer vision was used to automatically assign rewards during supervised learning



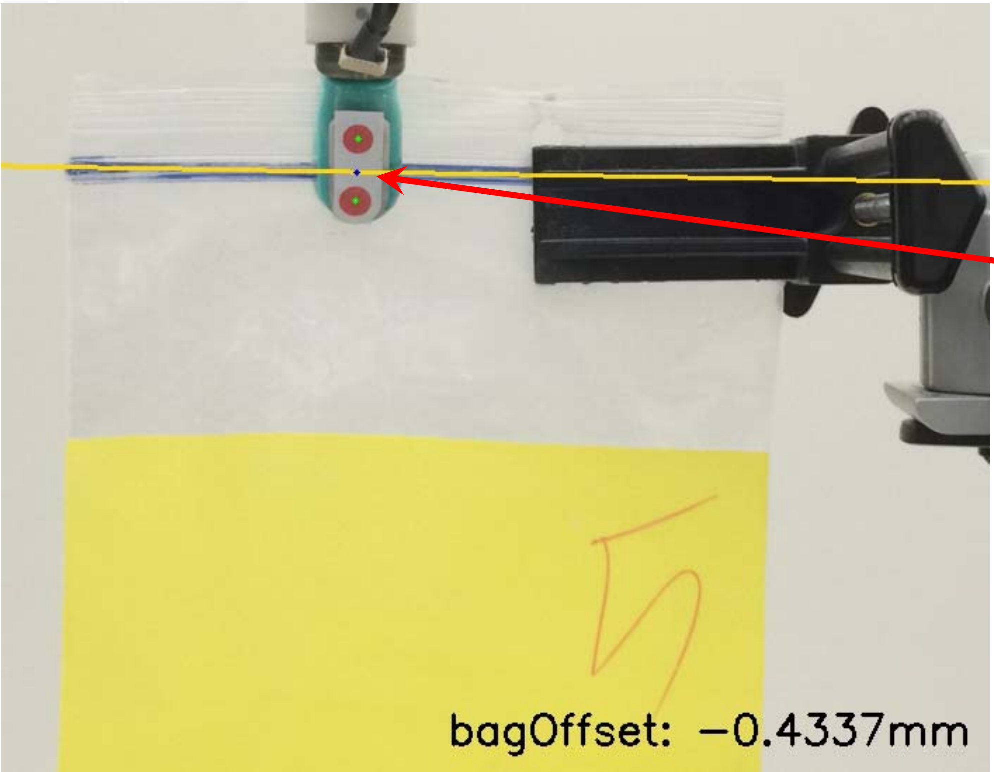
Green dots mark the centers of red circles placed over the fingernail screws

Computer vision was used to automatically assign rewards during supervised learning



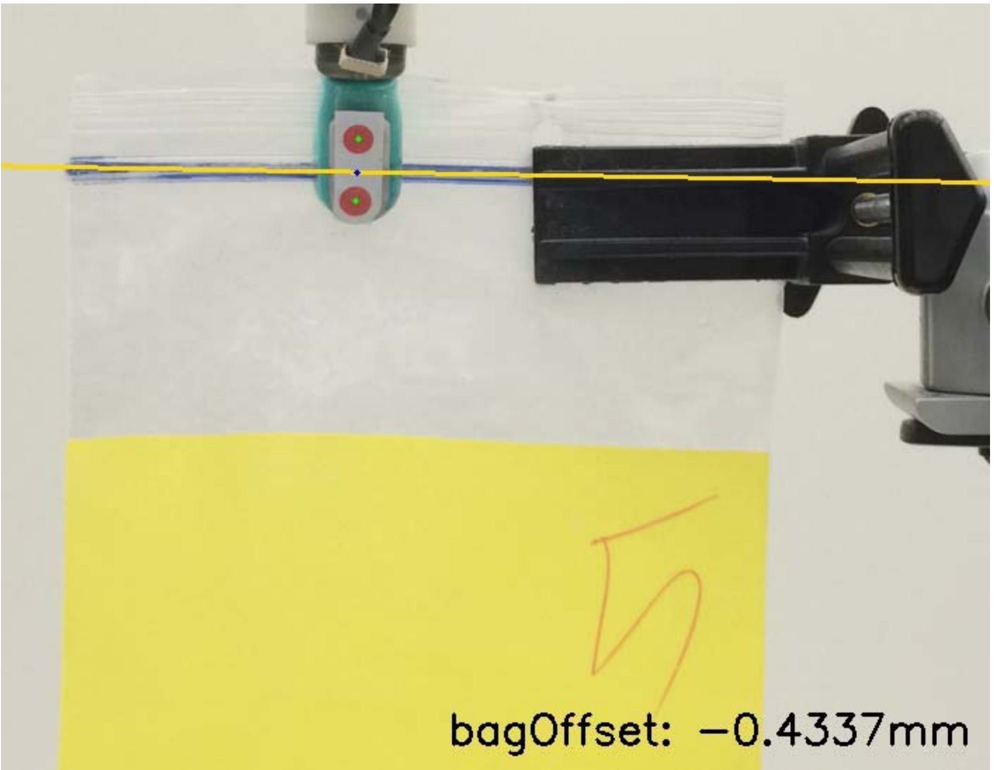
Yellow line marks the straight-line fit of the blue zipper

Computer vision was used to automatically assign rewards during supervised learning



Blue dot marks the estimated location of the zipper along the fingerpad

Computer vision was used to automatically assign rewards during supervised learning



Condition	Reward
0 mm < offset	0
-2.5 < offset ≤ 0	**+1**
offset ≤ -2.5 mm	0

Brief overview of C-MAB implementation

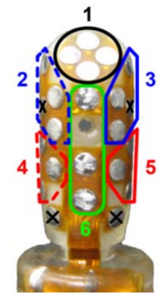
1. Send context (vector of tactile sensor data) to the DNN classifier, which returns a state label ("low," "center," "high").

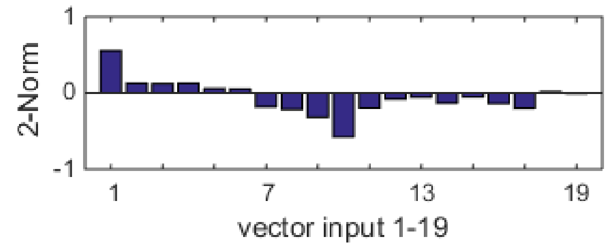
2. Calculate the control function $D(t) = t^{z} \ln(t)$ that depends on the similarity of the states and the size of the action space.

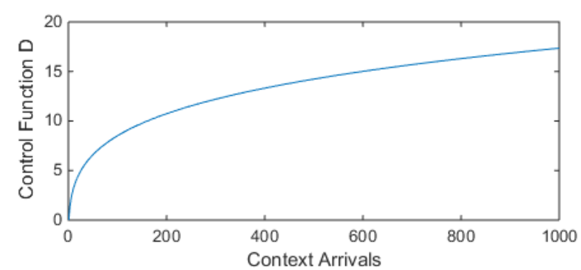
3. For the current state, check for underexplored actions by comparing state-action counts ("context arrivals") to $D(t)$.

4. If any counts are less than $D(t)$, execute an underexplored action at random. Otherwise, exploit the current policy.

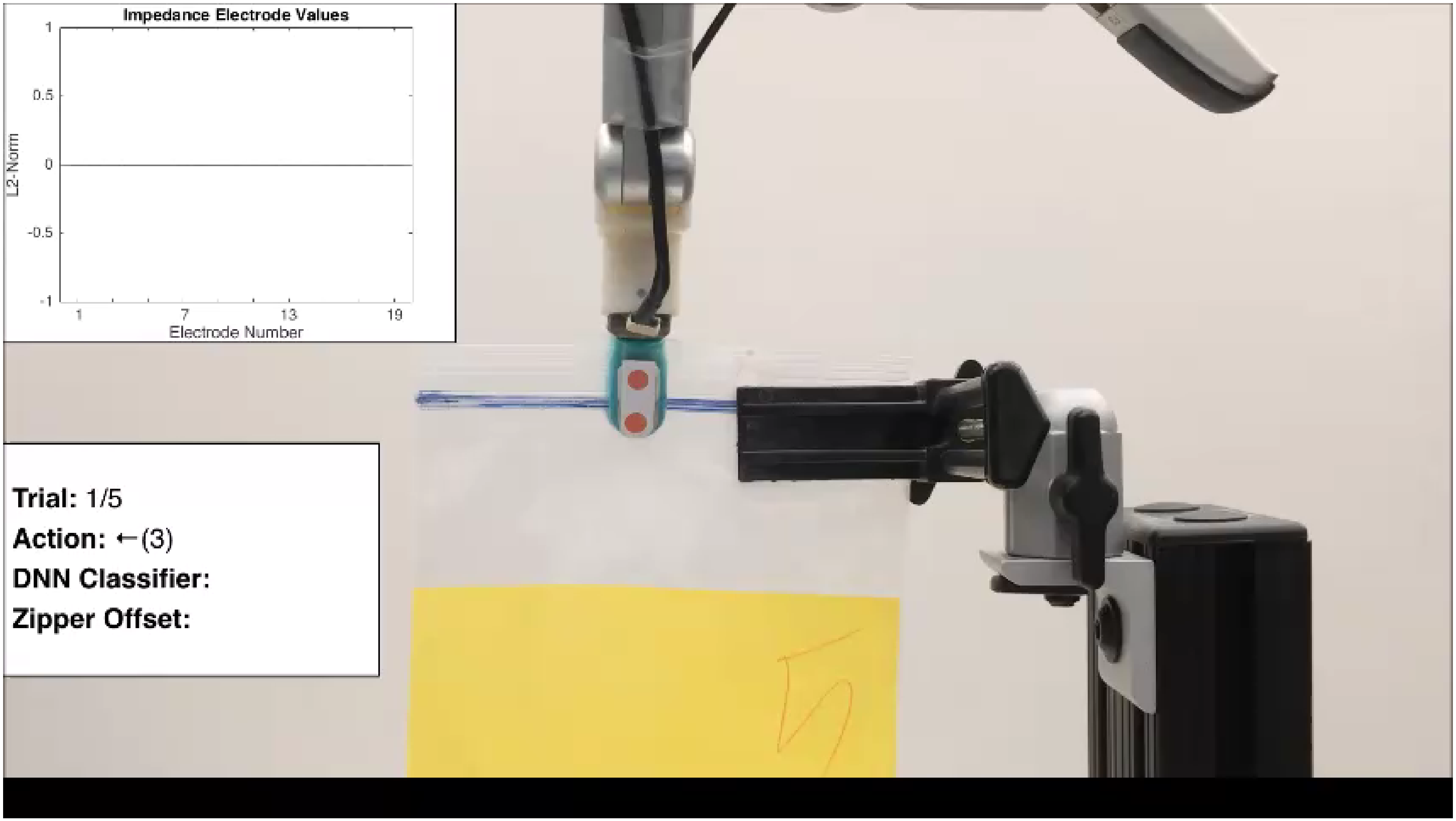
5. Update expected rewards and state-action counts.



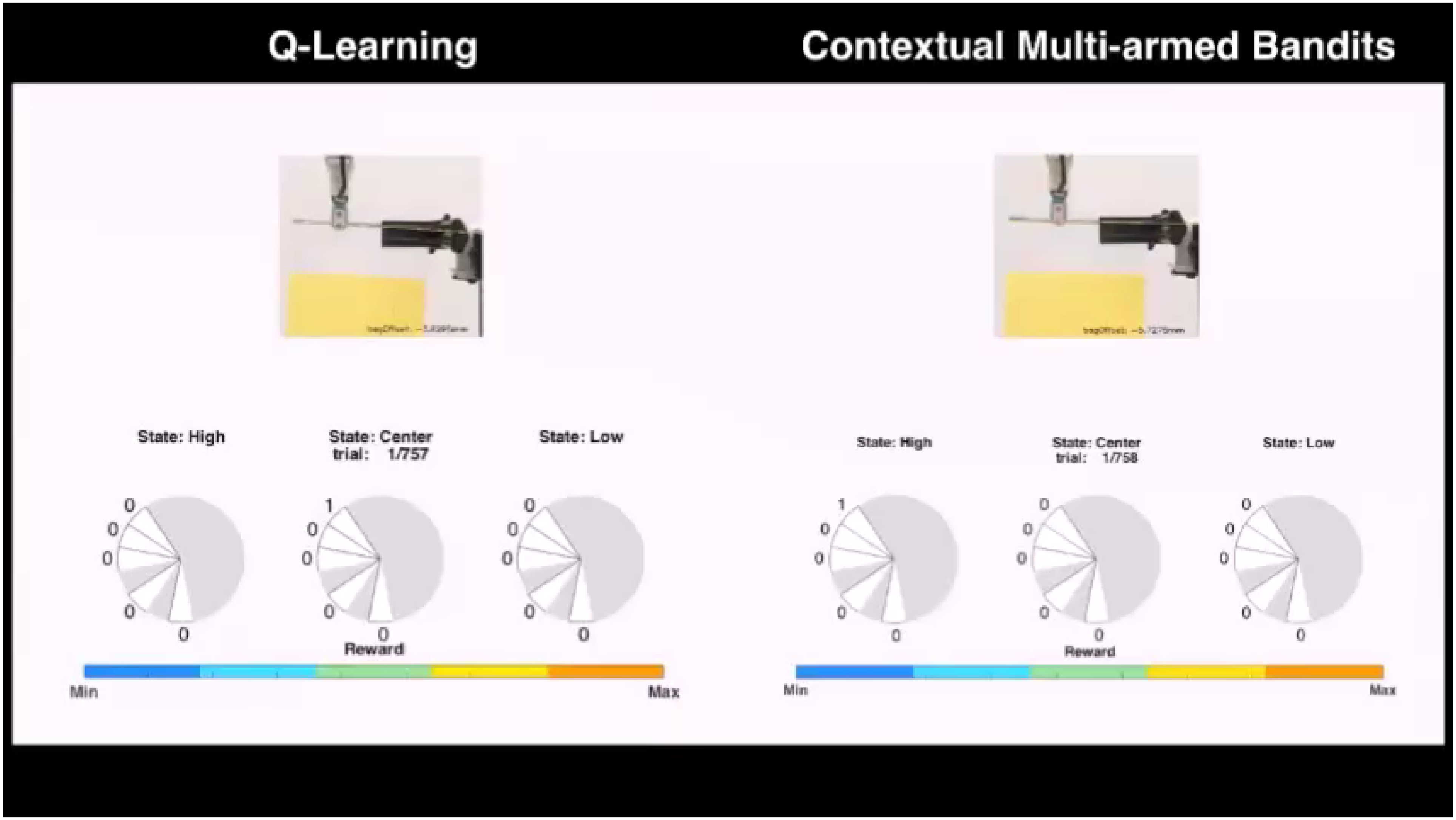




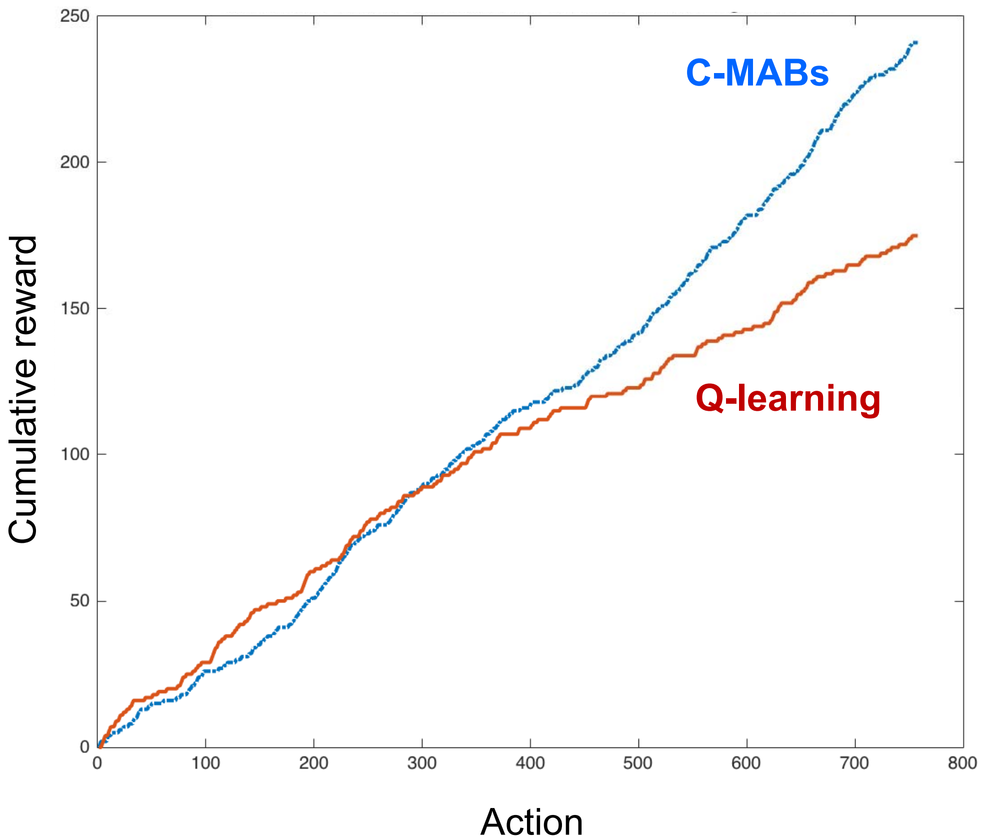
Online learning of expected rewards through exploration



Comparison of reinforcement learning algorithms



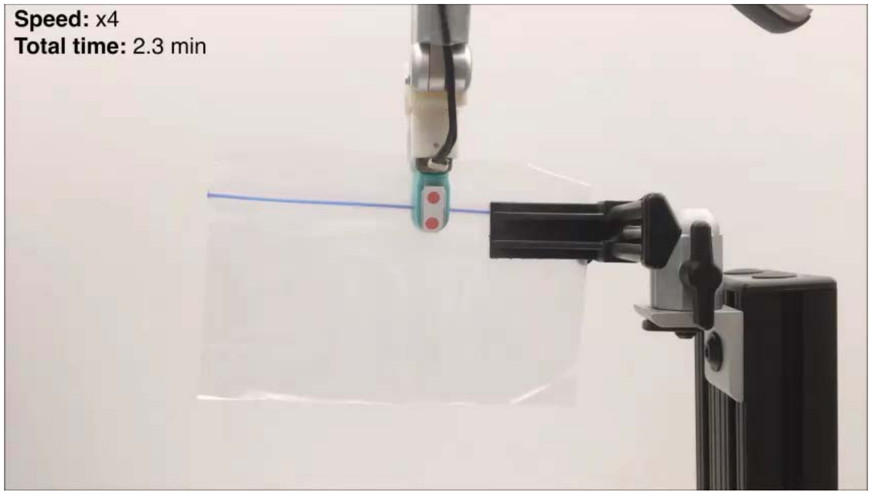
Comparison of cumulative rewards

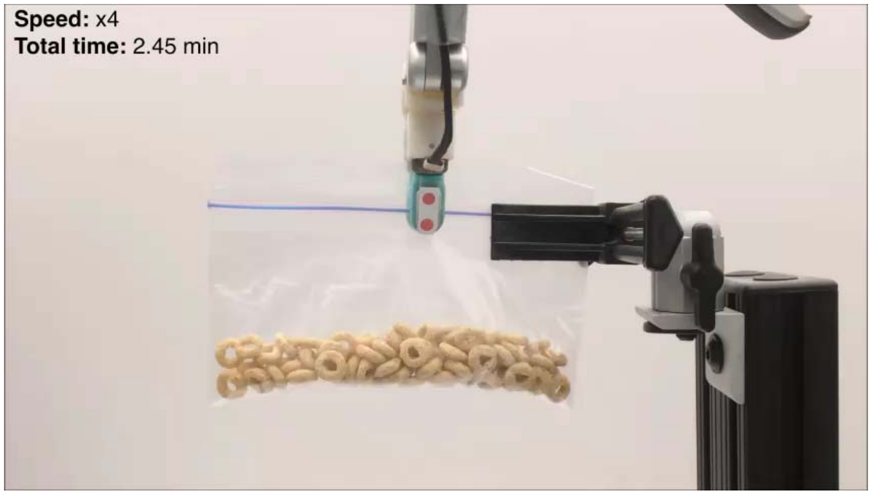


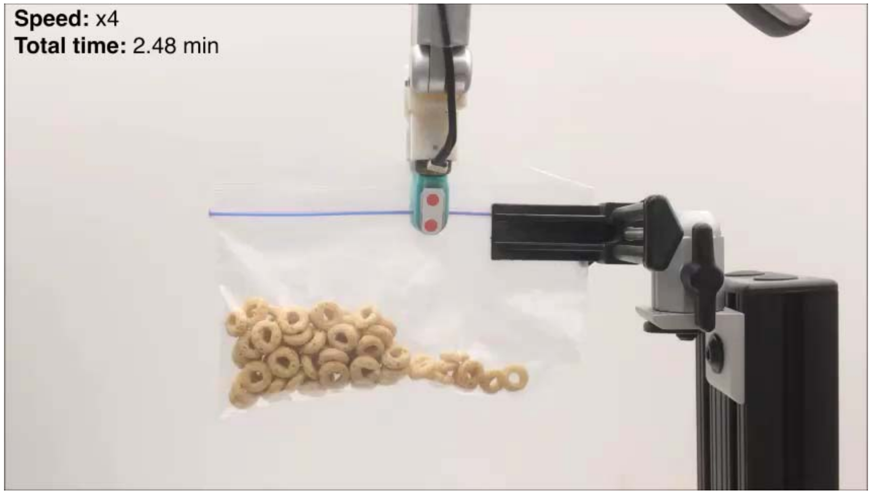
- Q-learning will converge to an optimal policy as time goes to infinity, but C-MABs outperform Q-learning within a finite number of trials.
- While the Q-learning parameters could be manually tuned to improve performance, manual tuning is avoided through the use of the more advanced C-MAB learner.

Testing the robustness of the C-MAB policy

Novel, more flexible ziplock bag under different loading conditions:



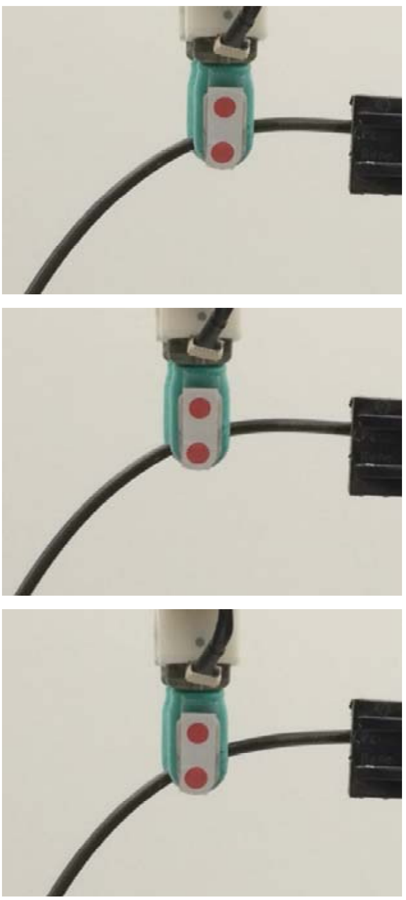




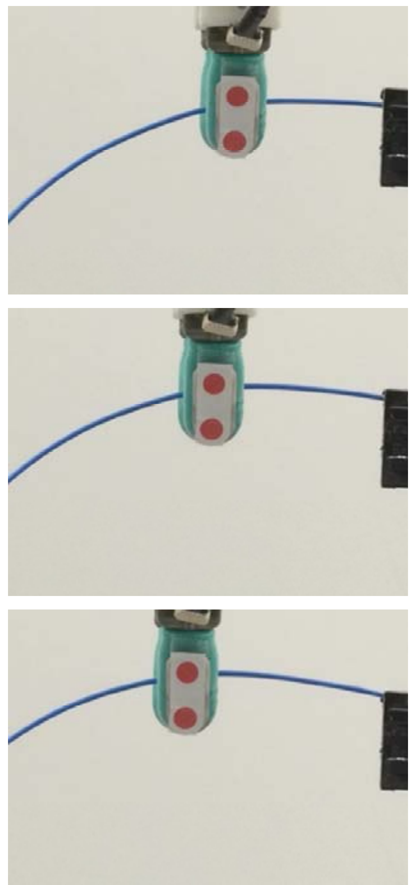
Testing the robustness of the C-MAB policy

Novel, deformable contours that were not zippers:

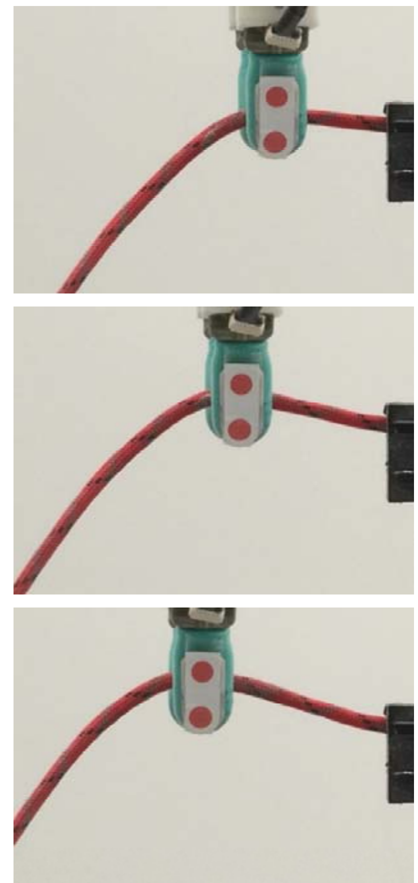
Thick electrical wire
(3.5 mm diam.)



Thin electrical wire
(1.5 mm diam.)

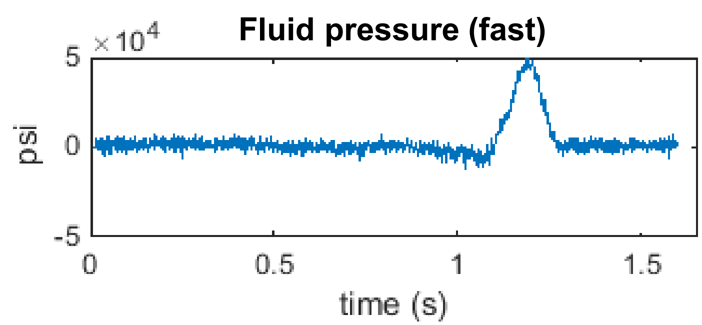


Nylon rope
(4 mm diam.)



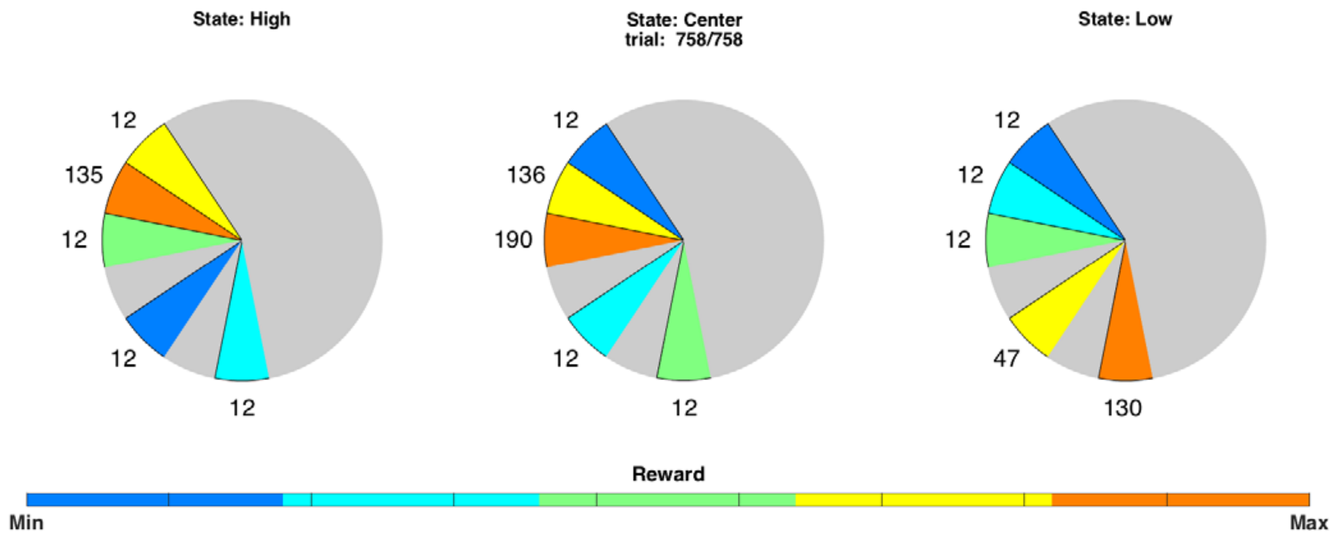
Potential future improvements

- Expand the action space
  - Online modulation of grasp pressure
  - Adjustments to fingertip travel length or velocity based on confidence
  - Out-of-plane movements and rotations of the fingertips
- Use adaptive algorithms to zoom in and refine regions of the state-action space with high context arrival counts.
- Reduce time delays due to 3D motion planning for the 7DOF robot arm through parallelized code and GPUs.
- Autonomously end the task using a haptic cue, such as the vibratory “click” upon zipper closure.



Discussion

- Tactile sensor data are difficult to simulate, time consuming to collect, and cause wear of the robot during collection. **Resource-conscious learning techniques are important** for the development of new complex skills that require repeated interactions between the robot and the environment.
- The learned C-MAB policy makes physical sense, but is not what we would have naively coded. **Non-intuitive solutions can be found by exploring** the state-action space.

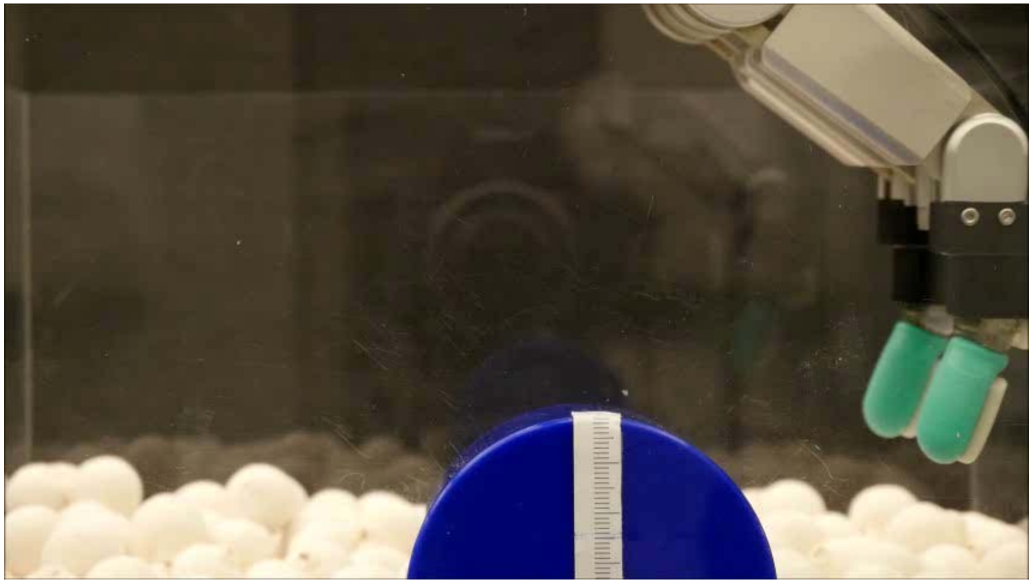


Haptic perception within granular media

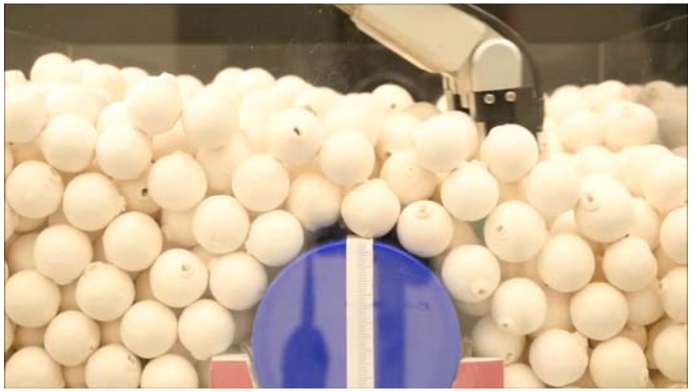
Without sensors that see through matter, the sense of touch is essential for locating, identifying, and grasping buried objects.

Image from (Hoffman, 2014).

Challenge: Granular media can make haptic perception difficult.

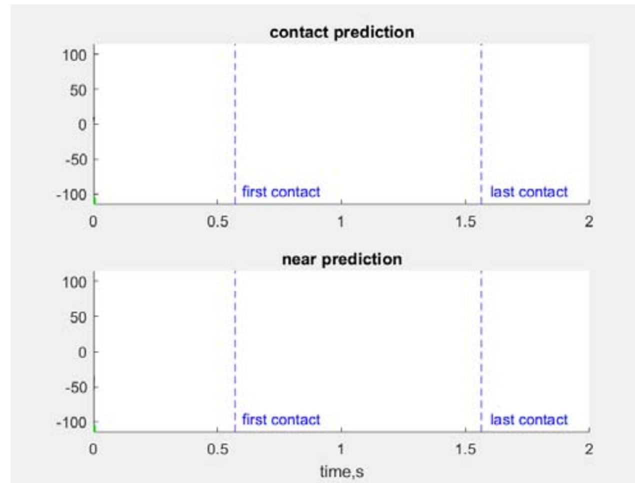


Sparse, overcomplete feature learning of tactile sensor data

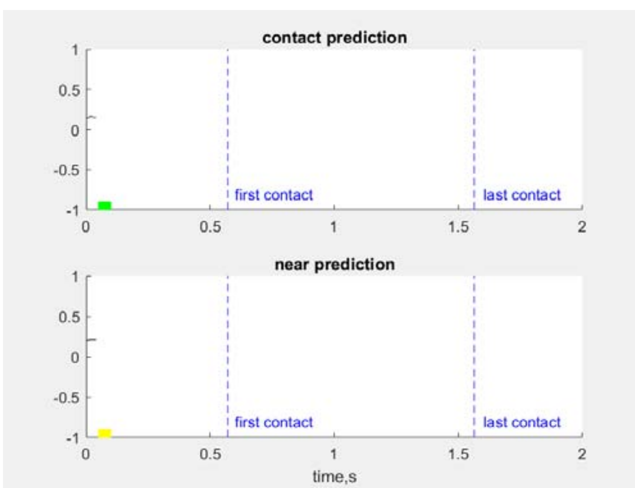


no object nearby, object nearby, contact with object

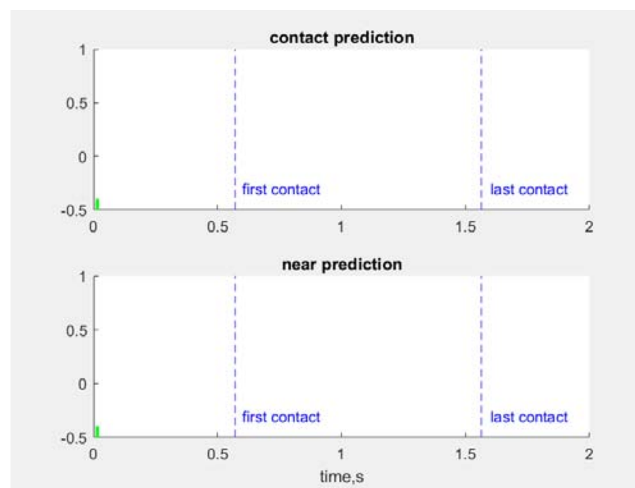
Fluid pressure (fast)



Fluid pressure (slow)



Electrode impedance

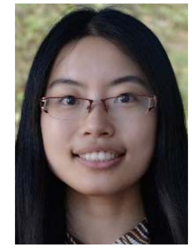


Acknowledgments

Dr. Randall Hellman
Mech. Engin.

Dr. Cem Tekin
Bilkent Univ.
Elec. Engin.

Dr. Mihaela van der Schaar
UCLA EE



Shengxin Jia,
PhD student,
Mech. Engin.

Members of the Biomechatronics Lab
BiomechatronicsLab.ucla.edu