Principles of Robot Autonomy II

Learning-based Approaches to Manipulation & Interactive Perception

Jeannette Bohg
Learning Outcome for next four Lectures

Modeling and Evaluating Grasps

Modeling and Executing Manipulation

Apply Learning to Grasping and Manipulation

Use Manipulation to Perceive better
Today’s itinerary

• Recap
  • Learning-based Grasping

• Learning-based Approaches to
  • Planar Pushing
  • Contact-Rich Manipulation Tasks

• Interactive Perception
  • Perception is not passive but active
  • Not seeing, but looking
For a Deeper Dive into Grasping and Manipulation

• CS326 – Topics in Advanced Robotic Manipulation – Fall 2022
Data-Driven Approaches to Grasping


Covered up till now

This lecture

Grasping previously unseen objects using only 2D images without 3D meshes
Data collection

Solution? Use synthetic data!

Realistic rendering using ray tracing.
Enables automatic labeling: random lighting, color, orientation, size...
Image preprocessing

Edge filters (Y):

Texture filters (Y):

Average filter (Cb/Cr):

6 (edge) + 9 (texture) + 1 (average) * 2 =

17 features per patch
Hardware setup

5 dof arm

Random object location on uncluttered table top
Using more sensing modalities and data to learn features and grasp policies

- DexNet 1.0 – 4.0 – Berkeley – AutoLab
- Google Arm Farm

"Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection" by Levine et al. IJRR 2017.

## Conclusion: Two Approaches

<table>
<thead>
<tr>
<th></th>
<th>Dex-Net</th>
<th>Arm Farm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Setup</strong></td>
<td>Single object in simulation</td>
<td>Bin of objects in real world</td>
</tr>
<tr>
<td><strong>Number Data Points</strong></td>
<td>13,000 objects, 2.5M grasps</td>
<td>1,100 objects, 1.7M grasps</td>
</tr>
<tr>
<td><strong>Data Point</strong></td>
<td>(object, grasp, label = probability of success)</td>
<td>(Image, motor command, label = ground truth success)</td>
</tr>
<tr>
<td><strong>Diversity of Objects</strong></td>
<td>Rigid, Opaque</td>
<td>Rigid &amp; deformable, opaque &amp; translucent</td>
</tr>
<tr>
<td><strong>Object Representation</strong></td>
<td>3D Mesh Model</td>
<td>None</td>
</tr>
<tr>
<td><strong>Data Collection Method</strong></td>
<td>Generated in simulation</td>
<td>Self-supervised on real hardware</td>
</tr>
<tr>
<td><strong>Type of Learning</strong></td>
<td>Deep learning, reinforcement learning</td>
<td>End-to-end deep learning</td>
</tr>
</tbody>
</table>
Today’s itinerary

• Recap
  • Learning-based Grasping

• Learning-based Approaches to
  • Planar Pushing
  • Contact-Rich Manipulation Tasks

• Interactive Perception
  • Perception is not passive but active
  • Not seeing, but looking
Controlling through contact

Better Models
Better Feedback
Predictive Model

Sensory Observations

Model

Predicted Effect

Action

Model–Predictive Control
Modeling Planar Pushing

**Friction limit surface:** describes friction forces occurring when part slides over support.

When pushed with a wrench within the limit surface: **no motion**.

For **quasi-static pushing:** wrench on the limit surface; object twist normal to limit surface.

Relation between wrench cone, limit surface and unit twist sphere. Adopted from Chapter 37, Fig 37.10 in Springer Handbook of Robotics.
Validating Planar Pushing Models

IROS 2016. "More than a Million Ways to Be Pushed: A High-Fidelity Experimental Dataset of Planar Pushing" by Peter Yu and Maria Bauza.
Predicting physical effect
Keep Predicting
Bias-Variance Tradeoff

Error

Bias

Variance

Min Error

Model Complexity
Model-based
Bias-Variance Tradeoff

Model-based

Model Complexity

Error

Bias

Variance

Min Error
Data-Driven
Bias-Variance Tradeoff

Error

Learning

Min Error

Structure

Bias

Variance

Model-based

Model Complexity

Data-Driven
Hypothesis

**Physics Models** + **Learning** = **Generalization**

Example:

- Real motion
- Simulated motion (Lynch 1992)

BIG difference!
Hybrid Model

Sensory Observations → Learned Model → Parameters → Physics-based Model → Predicted Effect

End-to-End Loss on Effect
Testing Hypothesis on a Case Study

Analytic Model

Relation between wrench cone, limit surface and unit twist sphere. Adopted from Chapter 37, Fig 37.10 in Springer Handbook of Robotics.
Compared Architectures

Raw Sensory Observations

**Neural** Network only

- Action
- Sensory Observations
- Learned Model
- Predicted Effect

**Hybrid** Model

- Action
- Sensory Observations
- Learned Model
- Parameters
- Physics-based Model
- Predicted Effect

**Training**: End-to-End

**Loss**: Error between Predicted and Ground Truth Effect
Advantages

Data Efficiency    Extrapolation
Testing Data Efficiency

\[ \text{trans} \% \]

\[ \text{thousand training examples} \]

--- physics ---
Testing Data Efficiency

![Graph showing trans [%] vs. thousand training examples]

- Blue line: hybrid
- Dotted line: physics
Testing Data Efficiency

Training = Test Distribution

Graph showing the relationship between training and test data efficiency with respect to the number of training examples for neural, hybrid, and physics models.
Testing Generalization

New Pushing Angles & Contact Points

Interpolation

New Push Velocities

Extrapolation

New Object Shapes

Generalization to new push velocities

Extrapolation

---

\[
\text{trans} \times [\%] \quad \text{rot} \times [\%]
\]

\[
push \text{ velocity } [\text{mm/s}]
\]

---

\[
\cdots \quad \text{physics}
\]
Generalization to new push velocities

Extrapolation

- Training
- Testing

Graphs showing the transition and rotation percentages as a function of push velocity in mm/s, comparing physics and hybrid models.
Generalization to new push velocities

![Diagram showing training, testing, and extrapolation]

Extrapolation

Training ≠ Test Distribution

Graphs showing the performance of different methods (physics, neural, hybrid) across push velocities.

- trans [%]
- rot [%]
- push velocity [mm/s]

Legend:
- physics
- neural
- hybrid
Don’t throw away structure
Learn to extract given state representation from raw data
A Concrete Suggestion
Interpretability
Today’s itinerary

- Recap
  - Learning-based Grasping
- Learning-based Approaches to
  - Planar Pushing
  - Contact-Rich Manipulation Tasks
- Interactive Perception
  - Perception is not passive but active
  - Not seeing, but looking
Models will never be perfect
Human multimodal sensor–motor coordination

Sensory Inputs

Example of Task–Relevant Information

pose  geo  force  friction
Human multimodal sensor–motor coordination

Sensory Inputs

pose  geo  force  friction

Example of Task–Relevant Information

Action

π
Human multimodal sensor–motor coordination

Example of Task–Relevant Information

Sensory Inputs

pose geo force friction

Action

π
Generalizable representations for multimodal inputs
Generalizable representations for multimodal inputs

\[ \pi(f(o_1, o_2, o_3 \ldots o_n)) = a \]

multimodal sensory inputs

Learn representation

learn policy for new task instances

pose  geo  force  friction  ...
Experimental setup

Multimodal sensory inputs

RGB

force/torque

robot states
Experimental setup

Peg geometry

Training

Testing
Learning generalizable representation

Inputs
- RGB image
- Force data
- Robot state

Encoder

Decoder

Representation
Learning generalizable representation

Inputs
- RGB image
- Force data
- Robot state

Fusion

Representation

Decoder
Learning generalizable representation

Inputs
- RGB image
- Force data
- Robot state

Fusion

Representation

Decoder

Task-relevant labels
- pose
- geo
- friction

EXPENSIVE
Collect labeled data with self-supervision
Dynamics prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Encoder

Representation

Decoders
- Action-conditional optical flow

Robot action
Dynamics prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Encoder

Robot action

Decoders
- Action-conditional optical flow
- 0 / 1 contact in the next step?

Representation
Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Encoder
- robot action

Representation
- 0 / 1 contact in the next step?
- 0 / 1 time-aligned?

Decoders
- Action-conditional optical flow
Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Encoder

Representation
- robot action

Decoders
- Action-conditional optical flow
- 0 / 1 contact in the next step?
- 0 / 1 time-aligned?
Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Robot state 0 / 1 contact in the next step?

Action-conditional optical flow

Encoder

Representation

Decoders
- Action-conditional optical flow
- yes, time aligned

Diagram:
- Inputs
- Encoder
- Representation
- Decoders
Concurrency prediction from self-supervision

Unpaired Inputs
- RGB image
- Force data
- Robot state

Encoder
- robot action

Representation

Decoders
- Action-conditional optical flow
- 0 / 1 contact in the next step?
- 0 / 1 time-aligned?
Concurrency prediction from self-supervision

Unpaired Inputs
- RGB image
- Force data
- Robot state

Encoder
- robot action

Representation

Decoders
- Action-conditional optical flow
- 0 / 1 contact in the next step?
- no, not time aligned!
Learning sample efficient policies

Inputs
- RGB image
- Force data
- Robot state

Encoder

Representation
How to efficiently learn a policy?

- Inputs:
  - RGB image
  - Force data
  - Robot state

Encoder

Freeze 500k parameters

Representation

RL Policy
How to efficiently learn a policy?

Inputs
- RGB image
- Force data
- Robot state

Encoder

Freeze 500k parameters

Learn 15k parameters

RL Policy
We evaluate our representation with policy learning

Episode 0
0% success rate

Episode 100
21% success rate
We efficiently learn policies in 5 hours

Episode 300

73% success rate

Episode 300

71% success rate

Episode 300

92% success rate
Our multimodal policy is robust against sensor noise

1. Force Perturbation
2. Camera Occlusion
Camera Occlusion

Agent View

x1
How is each modality used?

Simulation Results
(Randomized box location)

<table>
<thead>
<tr>
<th>Modality</th>
<th>Success Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Force Only</td>
<td>1</td>
</tr>
<tr>
<td>Image Only</td>
<td>49</td>
</tr>
<tr>
<td>Force &amp; Image</td>
<td>77</td>
</tr>
</tbody>
</table>

**Force Only**: Can’t find box

**Image Only**: Struggles with peg alignment

**Force & Image**: Can learn full task completion
Does our representation generalize to new geometries?
Does our representation generalize to new geometries?

92% Success Rate

Tested on
Representation
Policy
Does our representation generalize to new policies?

92% Success Rate

Tested on

Representation
Policy

62% Success Rate

Policy does not transfer

Tested on

Representation
Policy
Does our representation generalize?

92% Success Rate

Tested on

Representation

Policy

62% Success Rate

Tested on

Representation

Policy does not transfer

92% Success Rate

Tested on

Representation

Representation transfers
Overview of method

Self-supervised data collection

\( o_{RGB}, o_{force}, o_{robot} \)

100k data points
90 minutes

Representation learning

\( f(o_{RGB}, o_{force}, o_{robot}) \)

20 epochs on GPU
24 hours

Policy learning

\( \pi(f(\cdot)) = a \)

Deep RL
5 hours

Lessons Learned

1. **Self-supervision** gives us **rich** learning objectives

2. Representation that captures **concurrency** and **dynamics** can **generalize** across task instances

3. Our experiments show that multimodal representation leads to **learning efficiency** and **policy robustness**
State Representation - Physically Meaningful or Learned?
Today’s itinerary

• Recap
  • Learning-based Grasping

• Learning-based Approaches to
  • Planar Pushing
  • Contact-Rich Manipulation Tasks

• Interactive Perception
  • Perception is not passive but active
  • Not seeing, but looking
Interactive Perception

$S \times A \times t$

Sensory Data  Actions  Time

Control

Exploiting Multi-Modality

J. J. Gibson (1966) - The Senses considered as a Perceptual System.
Concurrency of Motion and Sensing

Accumulation over Time

Thanks to Octavia Camps at Northeastern University, Boston
Interactive Perception

Selfsupervised Learning of a Multimodal Representation

Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Encoding:
- Robot action
- Representation

Decoding:
- Action-conditional optical flow
- 0 / 1 contact in the next step?
- 0 / 1 time-aligned?

Interactive Perception

Exploiting RGB, Depth and Motion for Instance Segmentation
Selfsupervised Learning of a Multimodal Representation
Instance segmentation from a robotics viewpoint.
Optical Flow - Scene Flow

A Database and Evaluation Methodology for Optical Flow. Baker et al. IJCV. 2011

Pixel-wise Prediction

Computing Scene Flow

Scene Flow = Displacement of one 3D point $P_i^t$ over time

$$S_i = P_i^{t-1} - P_i^t$$

$P_i^{t-1} = r(P_i^t - X_k, Q_k) + X_k + T_k$

Motion-based Object Segmentation

pixels that move rigidly belong to one object.

Trajectory feature
\[ \xi_k = [X_k, X_k + T_k] \]

Clustering

Projection of centroid

Center of Object

Translation

Rotation

Radius Enclosing Sphere

Data Set of Frame Pairs

### Qualitative Results - Synthetic

<table>
<thead>
<tr>
<th>RGB Inputs</th>
<th>PD-flow</th>
<th>Scene Flow</th>
<th>Ground Truth</th>
<th>HOMC</th>
<th>Segmentation</th>
<th>Ground truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>frame $I^{t-1}$</td>
<td>frame $I^t$</td>
<td>OurC+vL+Rig</td>
<td>Ground Truth</td>
<td>HOMC</td>
<td>OurC+vL+Rig</td>
<td>Ground truth</td>
</tr>
</tbody>
</table>

Qualitative Results - Real

<table>
<thead>
<tr>
<th>RGB Inputs</th>
<th>Scene Flow</th>
<th>Segmentation</th>
</tr>
</thead>
<tbody>
<tr>
<td>frame $I^{t-1}$</td>
<td>PD-flow</td>
<td>HOMC</td>
</tr>
<tr>
<td>frame $I^t$</td>
<td>OurC+vL+Rig</td>
<td>OurC+vL+Rig</td>
</tr>
</tbody>
</table>

Interactive Perception

Exploiting RGB, Depth and Motion for Instance Segmentation

Interactive Perception

Homework 3 – Problem 3 Learning Intuitive Physics

Conclusions

• Interaction generates a rich sensory signal that eases perception.
• Action-conditional, multi-modal representation helps contact-rich manipulation and generalizes.
Suggested Reading

• "Data-Driven Grasp Synthesis – A survey" by Bohg et al. TRO 2014

• "Dex-Net 2.0: Deep Learning to Plan Robust Grasps with Synthetic Point Clouds and Analytic Grasp Metrics" by Mahler et al. RSS 2017. [https://berkeleyautomation.github.io/dex-net](https://berkeleyautomation.github.io/dex-net)

• "Learning Hand-Eye Coordination for Robotic Grasping with Deep Learning and Large-Scale Data Collection" by Levine et al. IJRR 2017.


• “Combining learned and analytical models for predicting action effects”, Kloss et al, IJRR 2020.

Next time

Guest Lecture by Karol Hausman

ABOUT

I’m a Senior Research Scientist at Google Brain and an Adjunct Professor at Stanford working on robotics and machine learning. I’m interested in enabling robots to autonomously acquire general-purpose skills with minimal supervision in real-world environments. I also co-teach a class at Stanford on Deep Multi-Task and Meta Learning (CS 330).

If you’re a Stanford student and you’d like to work together, please fill out this form, indicate your interest for us to work together and email me afterwards. Thanks!