Principles of Robot Autonomy II

Learning-based Approaches to Manipulation & Interactive Perception

Jeannette Bohg
Learning Outcome for next four Lectures

Modeling and Evaluating Grasps

Apply Learning to Grasping and Manipulation

Modeling and Executing 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 2023
Data-Driven Approaches to Grasping


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

Sees → Identifies 2d grasping point → Infers 3d grasping point → Instructs the arm to grasp the object
Data collection

Solution? Use synthetic data!

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

2500 images
5 object classes
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.

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
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

Model Complexity

Error

Bias

Variance

Min Error
Model-based
Bias-Variance Tradeoff

Error

Bias

Min Error

Model-based

Model Complexity

Variance

Min Error
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

Extrapolation

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**

- Sensory Observations
- Action
- Learned Model
- Predicted Effect

**Hybrid Model**

- Sensory Observations
- Action
- 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

graph showing the relationship between thousand training examples and trans [%]

the graph shows a horizontal line at 20%
Testing Data Efficiency

![Graph showing the efficiency of hybrid and physics methods over thousand training examples. The efficiency decreases consistently as the number of training examples increases. The hybrid method shows a consistently higher efficiency compared to the physics method.]}
Testing Data Efficiency

- Training = Test Distribution

Graph showing the relationship between training examples and the 'trans' variable, with lines for 'neural', 'hybrid', and 'physics' categories.
Testing Generalization

New Pushing Angles & Contact Points

New Push Velocities

New Object Shapes

Interpolation

Extrapolation

Generalization to new push velocities

Extrapolation
Generalization to new push velocities

Extrapolation

Graphs showing the performance of different methods (physics and hybrid) across various push velocities.
Generalization to new push velocities
Don’t throw away structure
Learn to extract given state representation from raw data
A Concrete Suggestion

Sensory Observations

Learned Model

Parameters

Physics-based Model

Predicted Effect

End-to-End Loss on Effect
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

Example of Task–Relevant Information

pose, geo, force, friction

Action

48
Human multimodal sensor–motor coordination

Sensory Inputs

Example of Task–Relevant Information

Action

pose geo force friction
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  ...

 Earn representation for new task instances
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
- Robot action
- Representation

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

Inputs
- RGB image
- Force data
- Robot state

Encoder
- robot action

Representation

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

Inputs
- RGB image
- Force data
- Robot state

Encoder

Representation

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

Robot action
Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

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

Diagram:
- Encoder
- Representation
- Decoders
- Robot action

Diagram elements:
- Inputs
- Outputs
- Encoder
- Decoders
- Robot action
- Representation
- 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

Encoder

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

Representation

Robot action

yes, 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?
- 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!

Robot state
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

Encode
- 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

Representation

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
Force Perturbation
How is each modality used?

Simulation Results (Randomized box location)

- **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

Policy does not transfer

Tested on

Representation

Policy

62% Success Rate

Tested on

Representation

Policy
Does our representation generalize?

92% Success Rate

Policy does not transfer

Tested on

Representation

Policy

62% Success Rate

Tested on

Representation

Policy

92% Success Rate

Representation transfers

Tested on

Representation

Policy
Overview of method

Self-supervised data collection

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

Representation learning

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

Policy learning

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

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?

Explicit Representation

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

Learned Representation

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

- Predicted Effect
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

Look  Rotate  Touch

Success Rate

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

Held and Hein (1963). Movement-Produced Stimulation in the Development of Visually-Guided Behaviour
Accumulation over Time

Thanks to Octavia Camps at Northeastern University, Boston
Interactive Perception

Selfsupervised Learning of a Multimodal Representation

Interactive Perception – Leveraging Action in Perception and Perception in Action.
Bohg, Hausman, Sanakaran, Brock, Kragic, Schaal and Sukhatme. TRO ’17.
Concurrency prediction from self-supervision

Inputs
- RGB image
- Force data
- Robot state

Decoder
- Action-conditional optical flow

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

 representation

Interactive Perception

Selfsupervised Learning of a Multimodal Representation

Exploiting RGB, Depth and Motion for Instance Segmentation

Instance segmentation from a robotics viewpoint.
Optical Flow - Scene Flow


Pixel-wise Prediction

Computing Scene Flow

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

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

Motion-based Object Segmentation

Pixels that move rigidly belong to one object.

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

Clustering

3D World Space at Frame t
Data Set of Frame Pairs

Qualitative Results - Synthetic

<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, OurC+vL+Rig</td>
<td>HOMC, OurC+vL+Rig</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 $T^{t-1}$</td>
<td>PD-flow</td>
<td>HOMC</td>
</tr>
<tr>
<td>frame $T^t$</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

S x A x t

Sensory Data  Actions  Time

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


• "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!