Navigating to Objects in the Real World

Theophile Gervet, Soumith Chintala, Dhruv Batra, Jitendra Malik, Devendra Singh Chaplot

Presented by: Jemuel Stanley Premkumar
Unseen environment: No experience, No map

Inputs
- Bedroom
- Goal category
- Observation
- \((x, y, \theta)\)
- Pose sensor

Output

Action
Spatial Scene Understanding
Navigable Space Detection

Semantic Scene Understanding
Object Detection

Semantic Exploration Priors
Where is a toilet more likely to be found?

Episodic Memory
Keep track of explored and unexplored areas
INTRODUCTION

Objective: Semantic Object Goal
Navigation
- Understanding of objects
- Likely location
- Exploit prior knowledge
WHY IS NAVIGATION CHALLENGING?

Poor spatial understanding

Poor semantic exploration priors

Poor semantic understanding

Poor episodic memory
HOW IS DONE NOW?

Classical
HOW IS DONE NOW?

End-to-end Learning

(x, y, θ)
Pose

Depth

RGB

Plant

Goal Category

Segment
Semantics

Resnet50 (PointNav)

Resnet18

Previous Action

Hidden State

GRU

Action

[Image: Habitat-Web: Learning Embedded Object-Search Strategies from Human Demonstrations et al., CVPR 2022]
HOW IS DONE NOW?

Modular Learning
How well do they perform?

**Classical**
- Geometric Map
- Heuristic Exploration
- No Training

**End-to-end Learning**
- End-to-end
- Large-scale IL + RL fine-tuning
  - 77,000 human trajectories
  - 200M frames of RL

**Modular Learning**
- Semantic Map
- Goal-Oriented Exploration
  - 10M frames of RL
Empirical Evaluation
3 Approaches
6 unseen homes
6 Global Object categories
Modular vs Classical

Goal: bed

SPL: 0.90, 98 steps
Semantic Exploration

SPL: 0.52, 152 steps
Frontier Exploration
End-to-end Learning Failure
RESULTS

➢ 200 Robot Actions

<table>
<thead>
<tr>
<th>Method</th>
<th>Success Rate</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modular Learning</td>
<td>0.81</td>
<td>0.47</td>
</tr>
<tr>
<td>Classical</td>
<td>0.78</td>
<td>0.42</td>
</tr>
<tr>
<td>End-to-end Learning</td>
<td>0.77</td>
<td>0.39</td>
</tr>
</tbody>
</table>
RESULTS

<table>
<thead>
<tr>
<th>Method</th>
<th>Success Rate</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Sim</td>
<td>Real World</td>
</tr>
<tr>
<td>Modular Learning</td>
<td>0.81</td>
<td>0.90</td>
</tr>
<tr>
<td>Classical</td>
<td>0.78</td>
<td>0.80</td>
</tr>
<tr>
<td>End-to-end Learning</td>
<td>0.23</td>
<td>0.78</td>
</tr>
</tbody>
</table>
How is **Modular Learning** better than **End-to-end Learning**?

Reconstructed one real-world home in simulation and conducted experiments with identical episodes in sim and reality.
How is **Modular Learning** better than **End-to-end Learning**?

**Sim vs Real**

**Sim**: Operates directly on RGB-D frames
- Domain gap

**Real**: Semantic map
- Invariant between sim and reality
How is **Modular Learning** better than **End-to-end Learning**?

**Error Modes**

**Success Rate**
- Sim
- Real World

- **Visual Reconstruction Errors** (including Segmentation Errors)
  - Sim: 0.81
  - Real World: 0.90

- **Physical Reconstruction Errors**
  - Sim: 0.79
  - Real World: 1.00

- **Depth Sensor Errors**
  - Exploration Failures
How is **Modular Learning** better than **End-to-end Learning**?

Error Modes

- Door approach at an angle
- Noisy depth
- Closed door
- Mirror reflection
- Reflected depth
- Hallucinated bed mapped
- Collisions in mirror

Reconstructed TV
**Key Takeaways**

<table>
<thead>
<tr>
<th>Practitioners</th>
<th>Researchers</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓ Modular learning more reliable (90% success)</td>
<td></td>
</tr>
</tbody>
</table>

**Issues:**

- **Sim-to-real gap:**
  - Leverage modularity and abstraction in policies
  - Existence of disconnection between sim and real error modes
  - Evaluate semantic navigation on real robots
How to make Sim better?
- Data augmentation techniques: Adding noise, varying light conditions
- Photo realism?
Discussion

➢ Develop real-world error modes for simulators
  - Limits usefulness of sim to diagnose bottlenecks
  - Modeling occlusion, sensor noise
Discussion

➢ Design policies that can be transferred from sim to real
  - Prioritize real-world transfer
  - Replace policy architectures that directly operate on RGB-D with ones leveraging abstractions as common practices in other domains
  - Avoid training a segmentation model on sim data if policy architecture does not allow easy swapping
Discussion

➢ Abstractions

Oliver A Wong (olivestw) 8 hours ago
I thought it was interesting that the modular approach performed better in the real world than in simulation. It seemed that some aspects of the simulations themselves were faulty. For example, segmentation failing because objects don’t look like they’re supposed to, or simulated hallways that are too narrow to pass through. It makes me curious if simulated datasets are checked for quality or if it is solely up to the researchers who create them.

Alan Van Omen 3 hours ago
I was also surprised that the modular approach actually had significantly better performance in the real-world. I imagine if they somehow fixed some of the issues they talked about in simulation, such as the reflection of the tv or mirror, than there would be better performance in the simulation.

But it seems to be a common theme in papers we have read, that it is very useful to extract a useful lower-dimensional representation from raw pixel data rather than try to learn directly in an end-to-end manner, as the latent representations will not be as susceptible to the large amount of noise and distractor data present in raw sensory inputs.

Nathaniel Chong (nychong) 1 hour ago
I agree with Alan in that latent representations and abstractions of real sensory input are more useful for model generalization, even to the real world.

The classical approach also performed better in the real world, indicating that the construction of a map and the use of semantic exploration are a reason for this success.

Rajiv Govindjee 3 hours ago
It seems to me that the issue is not inherently with end-to-end training, but rather that we don’t currently have good structures and model architectures that allow for these intermediate signals to be learned (stably). If we can learn these intermediate signals, it should theoretically be possible to achieve much better performance than introducing artificial bottlenecks that may be inefficient.

In an ideal architecture, a model can learn different submodules and pass information between them as needed; that information might contain some version of a semantic map (for example) without the engineer ever having to specify this. Presumably, the human brain is not genetically preprogrammed to pass around semantic maps. The human brain is, however, genetically programmed to have certain physical structures and to produce neurons with physical/chemical properties in different areas. That structure is what we seem to be lacking for many problems in embodied AI.

Giant transformer models appear to suffice for language and vision problems alone, but integrating them together for a task like visual navigation remains difficult.

Sawan Patel (sawanpa) 9 hours ago
I agree with Rajiv, it’s interesting how these characteristic features are very common and reproducible in the human brain but are not precisely replicated in embodied AI. The lack of a stable representation for intermediate signaling is certainly a limiting factor.
Discussion

➢ Domain adaptation