Learning to Explore using Active Neural SLAM

ICLR-20

Webpage: https://devendrachaplot.github.io/projects/Neural-SLAM
Code: https://github.com/devendrachaplot/Neural-SLAM

Devendra Singh Chaplot
Dhiraj Gandhi
Saurabh Gupta
Abhinav Gupta
Ruslan Salakhutdinov
Exploration
Exploration
Exploration
Exploration
Exploration
Exploration
Exploration

• How to efficiently explore an unseen environment?
  • Memory/Mapping: Where have you been?
  • State/Pose Estimation: Where are you now?
  • Planning: Where do you need to go?
Exploration

- How to efficiently explore an unseen environment?
  - Memory/Mapping: Where have you been?
  - State/Pose Estimation: Where are you now?
  - Planning: Where do you need to go?

- Limitations of end-to-end RL:
  - High sample complexity
  - Ineffective in large environments
Exploration

- How to efficiently explore an unseen environment?
  - Memory/Mapping: Where have you been?
  - State/Pose Estimation: Where are you now?
  - Planning: Where do you need to go?

- Limitations of end-to-end RL:
  - High sample complexity
  - Ineffective in large environments

- Our solution: Active Neural SLAM
  - Structured spatial representations
  - Hierarchical policies
  - Analytical planners
Active Neural SLAM: Overview
Active Neural SLAM: Overview

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Neural SLAM ($f_{SLAM}$)

Pose Estimate ($\hat{x}_t$)

Map ($m_t$)
Active Neural SLAM: Overview
Active Neural SLAM: Overview

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Neural SLAM ($f_{SLAM}$)

Pose Estimate ($\hat{x}_t$)

Global Policy ($\pi_G$)

Long-term goal ($g_l^t$)

Map ($m_t$)

Short-term goal ($g_s^t$)

$f_{Plan}$
Active Neural SLAM: Overview
Neural SLAM Module

- Conv-Deconv Neural Network
- Trained with supervised learning
- Learns explicit structured map and pose representations
Global Policy

- Convolutional Neural Network
- Trained with reinforcement learning
- Operates at a course time-scale
Local Policy

- Convolutional Neural Network
- Trained with imitation learning
- Operates at a fine time-scale
Neural SLAM Module
Neural SLAM Module

Sensor Pose Reading \((x_{t-1}')\)

Observation \((s_{t-1})\)

Sensor Pose Reading \((x_t')\)

Observation \((s_t)\)
Neural SLAM Module

Sensor Pose Reading ($x'_{t-1}$)

Observation ($s_{t-1}$)

Mapper ($f_{Map}$)

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Mapper ($f_{Map}$)
Neural SLAM Module

Sensor Pose Reading ($x_t'$)
Observation ($s_t$)

Mapper ($f_{Map}$)

Egocentric Proj. ($P_{ego}^t$)

Observation ($s_{t-1}$)
Sensor Pose Reading ($x_{t-1}'$)

Egocentric Proj. ($P_{ego}^{t-1}$)
Neural SLAM Module

Sensor Pose Reading \((x_t)\)

Observation \((s_t)\)

Sensor Pose Reading \((x_t')\)

Observation \((s_{t-1})\)

Mapper \((f_{Map})\)

Relative Pose Change \((d_x)\)

Egocentric Proj. \((p_{t-1}^{ego})\)

Mapper \((f_{Map})\)

Egocentric Proj. \((p_t^{ego})\)
Neural SLAM Module

Sensor Pose Reading \((x'_t)\)

Observation \((s_t)\)

Mapper \((f_{Map})\)

Relative Pose Change \((dx)\)

Egocentric Proj. \((p_{t}^{ego})\)

ST

Mapper \((f_{Map})\)

Egocentric Proj. \((p_{t-1}^{ego})\)
Neural SLAM Module

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Mapper ($f_{Map}$)

Relative Pose Change ($dx$)

Mapper ($f_{Map}$)

Egocentric Proj. ($p_t^{ego}$)

Egocentric Proj. ($p_{t-1}^{ego}$)

Observation ($s_{t-1}$)

Sensor Pose Reading ($x'_{t-1}$)

ST

Relative Pose Change ($dx_{t-1}$)
Neural SLAM Module

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Sensor Pose Reading ($x'_t$)

Observation ($s_{t-1}$)

Mapper ($f_{Map}$)

Relative Pose Change ($dx$)

Mapper ($f_{Map}$)

Egocentric Proj. ($p_t^{ego}$)

ST

Egocentric Proj. ($p_t^{ego}$)

Pose Estimate ($x_{t-1}$)

Pose Estimator ($f_{PE}$)

Relative Pose Change ($dx$)

Observation ($s_{t-1}$)
Neural SLAM Module

Sensor Pose Reading ($x_{t-1}$)

 Observation ($s_{t-1}$)

Mapper ($f_{Map}$)

Relative Pose Change ($dx$)

Egocentric Proj. ($P_{t-1}^{ego}$)

ST

Pose Estimator ($f_{PE}$)

Pose Estimate ($\hat{x}_t$)

Mapper ($f_{Map}$)

Egocentric Proj. ($P_t^{ego}$)

Pose Estimate ($\hat{x}_{t-1}$)

Observation ($s_t$)

Sensor Pose Reading ($x_t'$)
Neural SLAM Module

Sensor Pose Reading ($x'_t$)
Observation ($s_{t-1}$)

Mapper ($f_{Map}$)
Relative Pose Change ($dx$)

Egocentric Proj. ($p_{t-1}^{ego}$)

Mapper ($f_{Map}$)

Pose Estimate ($\hat{x}_t$)

Pose Estimator ($f_{PE}$)

Egocentric Proj. ($p_t^{ego}$)

ST

Pose Estimate ($\hat{x}_{t-1}$)
Neural SLAM Module

Sensor Pose Reading \( (x'_{t-1}) \)

Observation \( (s_{t-1}) \)

Sensor Pose Reading \( (x'_t) \)

Observation \( (s_t) \)

Sensor Pose Reading \( (x'_t) \)

Observation \( (s_t) \)

Sensor Pose Reading \( (x'_{t-1}) \)

Observation \( (s_{t-1}) \)

Mapper \( (f_{Map}) \)

Egocentric Proj. \( (p_{t-1}^{ego}) \)

ST

Mapper \( (f_{Map}) \)

Relative Pose Change \( (dx) \)

Pose Estimator \( (f_{PE}) \)

Pose Estimate \( (\hat{x}_t) \)

Pose Estimate \( (\hat{x}_{t-1}) \)

Egocentric Proj. \( (p_t^{ego}) \)

Geocentric Proj. \( (p_t^{ego}) \)

ST
Neural SLAM Module

Sensor Pose Reading \( (x_{t-1}) \)

Observation \( (s_{t-1}) \)

Sensor Pose Reading \( (x_t) \)

Observation \( (s_t) \)

Mapper \( (f_{Map}) \)

Relative Pose Change \( (dx) \)

Mapper \( (f_{Map}) \)

Egocentric Proj. \( (p_t^{ego}) \)

Egocentric Proj. \( (p_{t-1}^{ego}) \)

ST

Pose Estimate \( (\hat{x}_{t-1}) \)

Pose Estimator \( (f_{PE}) \)

Pose Estimate \( (\hat{x}_t) \)

ST

Channel Pool

Map \( (m_{t-1}) \)

Map \( (m_t) \)

ST

Egocentric Proj. \( (p_t^{ego}) \)

Geocentric Proj. \( (p_t^{geo}) \)
Neural SLAM Module

Sensor Pose Reading \((x_{t-1}')\)

Observation \((s_{t-1})\)

Sensor Pose Reading \((x_t')\)

Observation \((s_t)\)

Mapper \((f_{Map})\)

Mapper \((f_{Map})\)

Relative Pose Change \((dx)\)

Relative Pose Change \((p_{t-1})\)

Egocentric Proj. \((p_{t-1}^{ego})\)

Egocentric Proj. \((p_t^{ego})\)

Pose Estimate \((\hat{x}_{t-1})\)

Pose Estimate \((\hat{x}_t)\)

Pose Estimate \((\hat{x}_{t-1})\)

Pose Estimate \((\hat{x}_t)\)

Mapper \((f_{Map})\)

Egocentric Proj. \((p_{t-1}^{ego})\)

Egocentric Proj. \((p_t^{ego})\)

ST

ST

Pose Estimator \((f_{PE})\)

Pose Estimator \((f_{PE})\)

Channel Pool

Map \((m_{t-1})\)

Map \((m_t)\)

Geocentric Proj. \((p_t^{ego})\)

Geocentric Proj. \((p_t^{ego})\)

Pose Estimate \((\hat{x}_t)\)
Neural SLAM Module

Sensor Pose Reading ($x'_t$)

Observation ($s_t$)

Observation ($s_{t-1}$)

Mapper ($f_{Map}$)

Relative Pose Change ($dx$)

Mapper ($f_{Map}$)

Egocentric Proj. ($p_{t-1}^{ego}$)

Egocentric Proj. ($p_t^{ego}$)

Egocentric Proj. ($p_{t-1}^{ego}$)

Egocentric Proj. ($p_t^{ego}$)

Pose Estimate ($\hat{x}_t$)

Pose Estimate ($\hat{x}_{t-1}$)

Pose Estimate ($\hat{x}_t$)

Pose Estimate ($\hat{x}_{t-1}$)

ST

Channel Pool

ST

Neural SLAM

Map ($m_{t-1}$)

Map ($m_t$)

Pose Estimate ($\hat{x}_t$)

Neural SLAM Geocentric Proj.

Pose Estimate ($\hat{x}_t$)
Exploration: Task Setup
Exploration: Task Setup


Exploration: Task Setup

- **Physical Realism:** Actuation and motion sensor noise models based on real-data

Exploration: Task Setup

- **Physical Realism:** Actuation and motion sensor noise models based on real-data
- **Objective:** Maximize the explored area
  - A cell is explored when it is known to be traversable
- **Metrics:**
  - **Coverage ($m^2$)** - absolute explored area or coverage
  - % **Coverage** - percentage of the environment explored

Exploration: Task Setup

- **Physical Realism**: Actuation and motion sensor noise models based on real-data
- Objective: Maximize the explored area
  - A cell is explored when it is known to be traversable
- Metrics:
  - **Coverage ($m^2$)** - absolute explored area or coverage
  - % **Coverage** - percentage of the environment explored
- Fixed episode length of 1000 steps

Exploration: Task Setup

- **Physical Realism:** Actuation and motion sensor noise models based on real-data
- Objective: Maximize the explored area
  - A cell is explored when it is known to be traversable
- Metrics:
  - **Coverage** ($m^2$) - absolute explored area or coverage
  - % **Coverage** - percentage of the environment explored
- Fixed episode length of 1000 steps
- All methods trained for 10 million frames

Demo Video: Exploration

Observation

Predicted Map and Pose

https://youtu.be/tlyz68j_jvE
Demo Video: Exploration

 observation

 predicted map and pose

https://youtu.be/tlyz68j_jvE
Exploration Results
Exploration Results

% Coverage

RL + 3LConv [1]
RL + Res18
RL + Res18 + AuxDepth [2]
RL + Res18 + ProjDepth [3]
Active Neural SLAM

Exploration Results

| RL + 3LConv [1] | 73.7 | 22.838 |
| RL + Res18     | 74.7 | 23.188 |
| RL + Res18 + ProjDepth [3] | 78.9 | 24.863 |
| Active Neural SLAM | 94.8 | 32.701 |

Exploration Results

<table>
<thead>
<tr>
<th>System</th>
<th>% Coverage</th>
<th>Coverage ($m^2$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RL + 3LConv [1]</td>
<td>73.7</td>
<td>22.838</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>74.7</td>
<td>23.188</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>78.9</td>
<td>24.863</td>
</tr>
<tr>
<td>Active Neural SLAM</td>
<td>94.8</td>
<td>32.701</td>
</tr>
</tbody>
</table>

Exploration Results

![Graph showing exploration results for different methods]

- **RL + 3LConv [1]**: 33.2\% Coverage, 47.758 m²
- **RL + Res18**: 34.1\% Coverage, 49.175 m²
- **RL + Res18 + AuxDepth [2]**: 35.6\% Coverage, 51.959 m²
- **RL + Res18 + ProjDepth [3]**: 37.8\% Coverage, 54.775 m²
- **Active Neural SLAM**: 52.1\% Coverage, 73.281 m²

# Ablation

<table>
<thead>
<tr>
<th>Method</th>
<th>Gibson % Cov.</th>
<th>Val Overall Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Val Large Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Val Small Cov. (m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANS w/o Local Policy + Det. Planner</td>
<td>0.941</td>
<td>32.188</td>
<td>0.845</td>
<td>53.999</td>
<td>0.980</td>
<td>23.464</td>
</tr>
<tr>
<td>ANS w/o Global Policy + FBE</td>
<td>0.925</td>
<td>30.981</td>
<td>0.782</td>
<td>49.731</td>
<td>0.982</td>
<td>23.481</td>
</tr>
<tr>
<td>ANS w/o Pose Estimation</td>
<td>0.916</td>
<td>30.746</td>
<td>0.771</td>
<td>49.518</td>
<td>0.973</td>
<td>23.237</td>
</tr>
<tr>
<td>ANS</td>
<td><strong>0.948</strong></td>
<td><strong>32.701</strong></td>
<td><strong>0.862</strong></td>
<td><strong>55.608</strong></td>
<td><strong>0.983</strong></td>
<td><strong>23.538</strong></td>
</tr>
</tbody>
</table>
# Ablation

<table>
<thead>
<tr>
<th>Method</th>
<th>Gibson Val Overall</th>
<th>Gibson Val Large</th>
<th>Gibson Val Small</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% Cov.</td>
<td>Cov. (m²)</td>
<td>% Cov.</td>
</tr>
<tr>
<td>ANS w/o Local Policy + Det. Planner</td>
<td>0.941</td>
<td>32.188</td>
<td>0.845</td>
</tr>
<tr>
<td>ANS w/o Global Policy + FBE</td>
<td>0.925</td>
<td>30.981</td>
<td>0.782</td>
</tr>
<tr>
<td>ANS w/o Pose Estimation</td>
<td>0.916</td>
<td>30.746</td>
<td>0.771</td>
</tr>
<tr>
<td>ANS</td>
<td>0.948</td>
<td><strong>32.701</strong></td>
<td><strong>0.862</strong></td>
</tr>
</tbody>
</table>

*Replace Local Policy by Analytical Deterministic Policy*
# Ablation

<table>
<thead>
<tr>
<th>Method</th>
<th>Gibson % Cov.</th>
<th>Gibson Val Overall Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Gibson Val Large Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Gibson Val Small Cov. (m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANS w/o Local Policy + Det. Planner</td>
<td>0.941</td>
<td>32.188</td>
<td>0.845</td>
<td>53.999</td>
<td>0.980</td>
<td>23.464</td>
</tr>
<tr>
<td>ANS w/o Global Policy + FBE</td>
<td>0.925</td>
<td>30.981</td>
<td>0.782</td>
<td>49.731</td>
<td>0.982</td>
<td>23.481</td>
</tr>
<tr>
<td>ANS w/o Pose Estimation</td>
<td>0.916</td>
<td>30.746</td>
<td>0.771</td>
<td>49.518</td>
<td>0.973</td>
<td>23.237</td>
</tr>
<tr>
<td>ANS</td>
<td><strong>0.948</strong></td>
<td><strong>32.701</strong></td>
<td><strong>0.862</strong></td>
<td><strong>55.608</strong></td>
<td><strong>0.983</strong></td>
<td><strong>23.538</strong></td>
</tr>
</tbody>
</table>

Replace Local Policy by Analytical Deterministic Policy

Replace Global Policy by Frontier-based Exploration
## Ablation

Local Policy does not improve much over deterministic policy

<table>
<thead>
<tr>
<th>Method</th>
<th>Gibson % Cov.</th>
<th>Val Overall Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Val Large Cov. (m²)</th>
<th>Gibson % Cov.</th>
<th>Val Small Cov. (m²)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANS w/o Local Policy + Det. Planner</td>
<td>0.941</td>
<td>32.188</td>
<td>0.845</td>
<td>53.999</td>
<td>0.980</td>
<td>23.464</td>
</tr>
<tr>
<td>ANS w/o Global Policy + FBE</td>
<td>0.925</td>
<td>30.981</td>
<td>0.782</td>
<td>49.731</td>
<td>0.982</td>
<td>23.481</td>
</tr>
<tr>
<td>ANS w/o Pose Estimation</td>
<td>0.916</td>
<td>30.746</td>
<td>0.771</td>
<td>49.518</td>
<td>0.973</td>
<td>23.237</td>
</tr>
<tr>
<td>ANS</td>
<td><strong>0.948</strong></td>
<td><strong>32.701</strong></td>
<td><strong>0.862</strong></td>
<td><strong>55.608</strong></td>
<td><strong>0.983</strong></td>
<td><strong>23.538</strong></td>
</tr>
</tbody>
</table>

Replace Local Policy by Analytical Deterministic Policy

Replace Global Policy by Frontier-based Exploration
## Ablation

<table>
<thead>
<tr>
<th>Method</th>
<th>Gibson Val Overall</th>
<th>Gibson Val Large</th>
<th>Gibson Val Small</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% Cov.</td>
<td>Cov. (m²)</td>
<td>% Cov.</td>
</tr>
<tr>
<td>ANS w/o Local Policy + Det. Planner</td>
<td>0.941</td>
<td>32.188</td>
<td>0.845</td>
</tr>
<tr>
<td>ANS w/o Global Policy + FBE</td>
<td>0.925</td>
<td>30.981</td>
<td>0.782</td>
</tr>
<tr>
<td>ANS w/o Pose Estimation</td>
<td>0.916</td>
<td>30.746</td>
<td>0.771</td>
</tr>
<tr>
<td>ANS</td>
<td><strong>0.948</strong></td>
<td><strong>32.701</strong></td>
<td><strong>0.862</strong></td>
</tr>
</tbody>
</table>

- **Replace Local Policy by Analytical Deterministic Policy**
- **Replace Global Policy by Frontier-based Exploration**

Local Policy does not improve much over deterministic policy

Global Policy and Pose Estimation mostly help in Large maps
Pointgoal: Task Transfer
Pointgoal: Task Transfer

- Objective: Navigate to goal coordinates
Pointgoal: Task Transfer

- Objective: Navigate to goal coordinates
- Metric: Success weighted by inverse Path Length (SPL)

\[
\frac{1}{N} \sum_{i=1}^{N} \text{Success} \times \frac{\text{ShortestPathLength}}{\text{PathLength}}
\]
Pointgoal: Task Transfer

- Objective: Navigate to goal coordinates
- Metric: Success weighted by inverse Path Length (SPL)
  \[ \frac{1}{N} \sum_{i=1}^{N} \text{Success} \times \frac{\text{ShortestPathLength}}{\text{PathLength}} \]
- Global Policy -> always gives the pointgoal as the long-term goal
Harder Datasets

- **Hard-GEDR**
  - Higher Geodesic to Euclidean distance ratio (GEDR)
  - Avg GEDR 2.5 vs 1.37, minimum GEDR is 2

- **Hard-Dist**
  - Higher Geodesic distance
  - Avg Dist 13.5m vs 7.0m, minimum Dist is 10m
PointGoal Results
PointGoal Results

Random
RL + Blind
RL + 3LConv [1]
RL + Res18
RL + Res18 + AuxDepth [2]
RL + Res18 + ProjDepth [3]
IL + Res18
IL + CMP [4]
Active Neural SLAM (ANS)
ANS + Task Transfer

*SPL

PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.021</td>
<td>0.027</td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.421</td>
<td>0.625</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.406</td>
<td>0.550</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.422</td>
<td>0.561</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.461</td>
<td>0.640</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.436</td>
<td>0.614</td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.725</td>
<td>0.823</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.73</td>
<td>0.827</td>
</tr>
<tr>
<td>Active Neural SLAM (ANS)</td>
<td>0.848</td>
<td>0.951</td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.846</td>
<td>0.950</td>
</tr>
</tbody>
</table>

PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.021</td>
<td>0.027</td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.421</td>
<td>0.625</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.406</td>
<td>0.550</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.422</td>
<td>0.561</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.461</td>
<td>0.640</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.436</td>
<td>0.614</td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.725</td>
<td>0.823</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.73</td>
<td>0.827</td>
</tr>
<tr>
<td>Active Neural SLAM (ANS)</td>
<td>0.848</td>
<td>0.951</td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.846</td>
<td>0.950</td>
</tr>
</tbody>
</table>

# PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.021</td>
<td>0.027</td>
</tr>
<tr>
<td>Reinforcement Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.421</td>
<td>0.625</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.406</td>
<td>0.550</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.422</td>
<td>0.561</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.461</td>
<td>0.640</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.436</td>
<td>0.614</td>
</tr>
<tr>
<td>Imitation Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.725</td>
<td>0.823</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.73</td>
<td>0.827</td>
</tr>
<tr>
<td>Active Neural SLAM (ANS)</td>
<td>0.848</td>
<td>0.951</td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.846</td>
<td>0.950</td>
</tr>
</tbody>
</table>

PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.021</td>
<td>0.027</td>
</tr>
<tr>
<td>Reinforcement Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.421</td>
<td>0.625</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.406</td>
<td>0.550</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.422</td>
<td>0.561</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.461</td>
<td>0.640</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.436</td>
<td>0.614</td>
</tr>
<tr>
<td>Imitation Learning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.725</td>
<td>0.823</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.73</td>
<td>0.827</td>
</tr>
<tr>
<td>Active Neural SLAM (ANS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ours</td>
<td>0.848</td>
<td>0.951</td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.846</td>
<td>0.950</td>
</tr>
</tbody>
</table>

Gibson

## PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td><strong>Reinforcement Learning</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.02</td>
<td>0.052</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.046</td>
<td>0.072</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.109</td>
<td>0.176</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.197</td>
<td>0.277</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.129</td>
<td>0.180</td>
</tr>
<tr>
<td><strong>Imitation Learning</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.558</td>
<td>0.682</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.553</td>
<td>0.670</td>
</tr>
<tr>
<td><strong>Active Neural SLAM (ANS)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.71</td>
<td>0.824</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td>0.703</td>
<td>0.821</td>
</tr>
</tbody>
</table>

# PointGoal Results

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0</td>
<td>0.000</td>
</tr>
<tr>
<td><strong>Reinforcement Learning</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.006</td>
<td>0.008</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.006</td>
<td>0.006</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.003</td>
<td>0.004</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.011</td>
<td>0.013</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.004</td>
<td>0.008</td>
</tr>
<tr>
<td><strong>Imitation Learning</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.31</td>
<td>0.359</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.318</td>
<td>0.369</td>
</tr>
<tr>
<td><strong>Active Neural SLAM (ANS)</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ANS + Task Transfer</td>
<td>0.534</td>
<td>0.662</td>
</tr>
<tr>
<td><strong>Ours</strong></td>
<td>0.532</td>
<td>0.665</td>
</tr>
</tbody>
</table>

PointGoal Results

SPL

<table>
<thead>
<tr>
<th>Method</th>
<th>SPL</th>
<th>Success</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>0.01</td>
<td>0.010</td>
</tr>
<tr>
<td>RL + Blind</td>
<td>0.087</td>
<td>0.136</td>
</tr>
<tr>
<td>RL + 3LConv [1]</td>
<td>0.08</td>
<td>0.102</td>
</tr>
<tr>
<td>RL + Res18</td>
<td>0.125</td>
<td>0.160</td>
</tr>
<tr>
<td>RL + Res18 + AuxDepth [2]</td>
<td>0.143</td>
<td>0.189</td>
</tr>
<tr>
<td>RL + Res18 + ProjDepth [3]</td>
<td>0.111</td>
<td>0.134</td>
</tr>
<tr>
<td>IL + Res18</td>
<td>0.318</td>
<td>0.365</td>
</tr>
<tr>
<td>IL + CMP [4]</td>
<td>0.27</td>
<td>0.320</td>
</tr>
<tr>
<td>Active Neural SLAM (ANS)</td>
<td>0.496</td>
<td>0.593</td>
</tr>
<tr>
<td>Ours ANS + Task Transfer</td>
<td>0.49</td>
<td>0.588</td>
</tr>
</tbody>
</table>

Results

Gibson

Domain Generalization
(Matterport 3D)

Goal Generalization
(Harder goals)

Exploration

Task Generalization

Pointgoal

https://youtu.be/tlyz68j_jvE

https://youtu.be/T2yfqrxC0Gg

https://youtu.be/4a3Mt7lmSK8

https://youtu.be/_k9r19qCcsk

Results

Gibson

Domain Generalization
(Matterport 3D)

Goal Generalization
(Harder goals)

Exploration

https://youtu.be/tlyz68j_jvE

https://youtu.be/T2yfqrxP0Gg

Task Generalization

Pointgoal

https://youtu.be/4a3Mt7lmSK8

https://youtu.be/_k9r19qCcsk

https://youtu.be/G6kc_GtItR8
Winner of CVPR 2019 Habitat Challenge

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active Neural SLAM (Arnold)</td>
<td>0.805</td>
</tr>
<tr>
<td>2</td>
<td>Mid-level-Features</td>
<td>0.800</td>
</tr>
<tr>
<td>3</td>
<td>CHROMA</td>
<td>0.712</td>
</tr>
<tr>
<td>4</td>
<td>ARF-RL</td>
<td>0.699</td>
</tr>
<tr>
<td>5</td>
<td>MTank</td>
<td>0.260</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active Neural SLAM (Arnold)</td>
<td>0.948</td>
</tr>
<tr>
<td>2</td>
<td>Pansy</td>
<td>0.927</td>
</tr>
<tr>
<td>3</td>
<td>Titardrew</td>
<td>0.868</td>
</tr>
<tr>
<td>4</td>
<td>Hiccup</td>
<td>0.846</td>
</tr>
<tr>
<td>5</td>
<td>CHROMA</td>
<td>0.843</td>
</tr>
</tbody>
</table>
## Winner of CVPR 2019 Habitat Challenge

### RGB Leaderboard

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active Neural SLAM (Arnold)</td>
<td>0.805</td>
</tr>
<tr>
<td>2</td>
<td>Mid-level-Features</td>
<td>0.800</td>
</tr>
<tr>
<td>3</td>
<td>CHROMA</td>
<td>0.712</td>
</tr>
<tr>
<td>4</td>
<td>ARF-RL</td>
<td>0.699</td>
</tr>
<tr>
<td>5</td>
<td>MTank</td>
<td>0.260</td>
</tr>
</tbody>
</table>

### RGBD Leaderboard

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team</th>
<th>SPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Active Neural SLAM (Arnold)</td>
<td>0.948</td>
</tr>
<tr>
<td>2</td>
<td>Pansy</td>
<td>0.927</td>
</tr>
<tr>
<td>3</td>
<td>Titardrew</td>
<td>0.868</td>
</tr>
<tr>
<td>4</td>
<td>Hiccup</td>
<td>0.846</td>
</tr>
<tr>
<td>5</td>
<td>CHROMA</td>
<td>0.843</td>
</tr>
</tbody>
</table>
Sim-to-Real Transfer

https://youtu.be/afqbn3gpeiA
Sim-to-Real Transfer

Observation

Predicted Map and Pose

https://youtu.be/afqbn3gpeiA
Learning to Explore using Active Neural SLAM

Webpage: https://devendrachaplot.github.io/projects/Neural-SLAM
Code: https://github.com/devendrachaplot/Neural-SLAM

Thank you

Devendra Singh Chaplot

Webpage: http://devendrachaplot.github.io/
Email: chaplot@cs.cmu.edu
Twitter: @dchaplot