Following High-level Navigation Instructions on a Simulated Quadcopter with Imitation Learning
Valts Blukis, Nataly Brukhim, Andrew Bennet, Ross A. Knepper and Yoav Artzi

Problem Statement

Goal: map instructions and visual observations to actions
Control a quadcopter to execute the instruction and stop at the goal location
Agent observes: first-person camera images and pose estimates.
Output: continuous velocity commands.

Go to the left side of the plane
Go to the front side of bin
Go to the back side of stump

Grounded Semantic Mapping Network (GSMN)

Inputs
- First-person image
- Pose
- Instruction

Perception
1. Extract a feature map with a ResNet Neural Network.
2. Project visual features with a pinhole camera model to the map reference frame.
3. Accumulate features in the semantic map over time.

Mapping
- 3x3 Conv
- 3x3 Conv
- 1x1 Conv
- Linear

Grounding
- Compute grounding map using a 1x1 language-derived convolutional filter.

Planning
- Compute goal map using a 9x9 language-derived filter.

Objective Function

Full Objective Function:
\[ J(\theta) = J_{\text{task}}(\theta) + \lambda_g J_{\text{percep}}(\theta) + \lambda_m J_{\text{lang}}(\theta) + \lambda_p J_{\text{ground}}(\theta) + \lambda_r J_{\text{plan}}(\theta) \]

Auxiliary Objectives
- Top-down view of environment
- Semantic Map
- Grounding Map
- Goal Map
- RNN

Imitation Learning with DAggerFM

- Learn by imitating actions given by oracle policy:
  1. Execute oracle and collect dataset D of trajectories, each a sequence of observations and ground-truth actions.
  2. Loop:
     2.1. Drop N trajectories from D.
     2.2. Execute agent policy to collect N trajectories and add to D. Every observation is annotated with ground-truth oracle actions.
     2.3. Update policy parameters by gradient descent given dataset D.

- Dataset D does not grow.
- Oracle is a simple carrot-planer control rule following ground truth trajectories.

Experimental Setup

- 3500/750/750 environments-instruction pairs for training/development/testing.
- 63 different objects, 6 to 13 per environment.
- Realistic-dynamics simulator based on Microsoft Airsim and Unreal Engine.

Results and Analysis

- Our model outperforms traditional neural architecture where instead of building a map, a recurrent network is used as memory.
- Almost reaches oracle performance.

- Resilient to noise in position estimates (0.5m std dev, map is 30x30m).
- Grounding auxiliary \( J_{\text{ground}}(\theta) \) is essential in our few-sample regime.
- Foal prediction auxiliary \( J_{\text{plan}}(\theta) \) is not essential for simple instructions.