The Surprising Effectiveness of Visual Odometry Techniques for Embodied PointGoal Navigation

Xiaoming Zhao, Harsh Agrawal, Dhruv Batra, Alexander Schwing

https://xiaoming-zhao.github.io/projects/pointnav-vo/

2021 ICCV OCTOBER 11-17 VIRTUAL

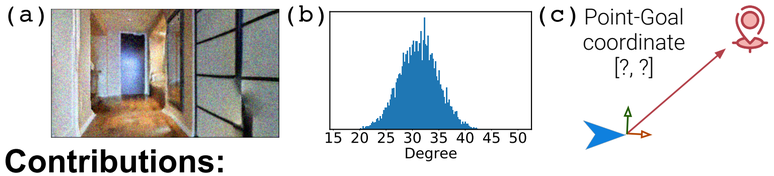
1. Motivation

Goal: Solve PointGoal navigation with realistic settings:

(a) Noisy egocentric sensors

(b) Realistic actuation (e.g., `turn_left` in figure below)

(c) No GPS or Compass data is accessible

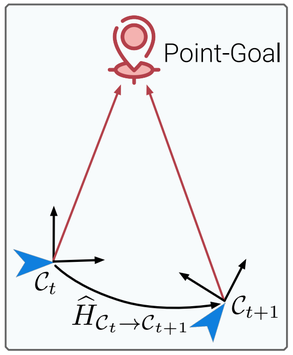


Contributions:

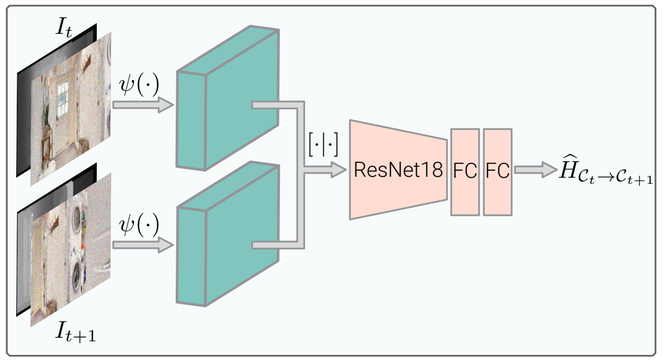
Verifying visual odometry techniques's effectiveness.

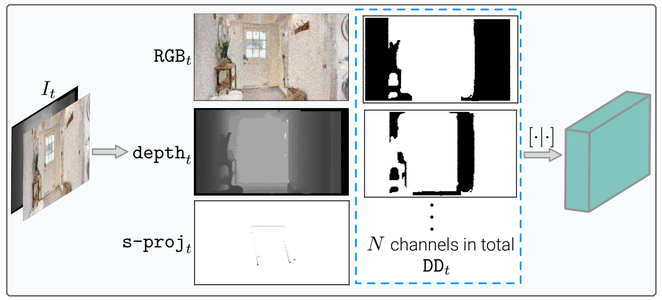
2. Overview

- We estimate agent's SE(2) transformation $\widehat{H}_{\mathcal{C}_t \to \mathcal{C}_{t+1}}$ when taking an action
- Given PointGoal's estimated position $\widehat{v}_t^g$, it is updated as
$$\widehat{v}_{t+1}^g = \widehat{H}_{\mathcal{C}_t \to \mathcal{C}_{t+1}} \cdot \widehat{v}_t^g$$
- Estimated PointGoal's position will be integrated into a navigation policy



3. Visual Odometry (VO) Techniques

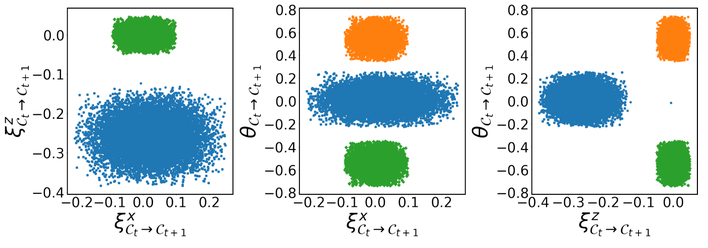




$[\cdot|\cdot]$: concatenate along channel dimension

VO module takes two adjacent RGB-D frames and outputs estimated SE(2) transformation. To increase robustness:

- **Regarding structure**:
  - Geometric invariance is encouraged
  - Action-specific design is chosen based on figure below
  - Dropout is added to last two FC layers
- **Regarding feature processing**:
  - Depth discretization DD_t is utilized
  - Egocentric top-down projection s-proj_t is employed



4. Results

Quantitatively: https://eval.ai/web/challenges/challenge-page/580/leaderboard/1631 as of 09/27/2021.

- Integrating VO into navigation policy achieves SOTA performance while executing 6.4 times faster
- Note, for row 1-2, we use VO as a **drop-in** module for a pre-trained navigation policy

Rank	Team	$S\uparrow$	SPL↑	$d_G\downarrow$	SoftSPL↑	Time $(h)\downarrow$
1-1	`Ours w/ finetuning`	**71.7**	**52.5**	0.802	**66.5**	5.83
1-2	`Ours w/o finetuning`	**69.8**	**52.0**	0.823	**65.7**	6.63
2	Karkus *et al.* 2021	64.5	37.7	**0.697**	52.1	37.50
3	Ramakrishnan *et al.* 2020	29.0	22.0	2.567	47.3	11.06
4	Information Bottleneck	16.3	12.2	2.075	56.1	2.73
5	Datta *et al.* 2020	15.7	11.9	2.232	58.6	**2.31**
6	cogmodel_team (39)	1.3	0.9	4.879	30.4	5.47
7	cso	1.2	0.7	4.632	24.7	5.57
8	UCULab	0.8	0.5	6.555	10.4	15.12
9	`Habitat Team`	0.3	0.0	6.929	3.8	-

Qualitatively: Agent is asked to navigate from blue square to green square (a failure case is illustrated in last figure).

- Blue curve is the actual path the agent takes
- Red curve is visualization of its location from the VO model by integrating over SE(2) estimation of each step