Towards Accurate Active Camera Localization

Qihang Fang[1*], Yingda Yin[2*], Qingnan Fan[3†], Fei Xia[4], Siyan Dong[1], Sheng Wang[5], Jue Wang[3], Leonidas Guibas[4], Baoquan Chen[2†]

[1]Shandong University, [2]Peking University, [3]Tencent AI Lab, [4]Stanford University, [5]3vjia

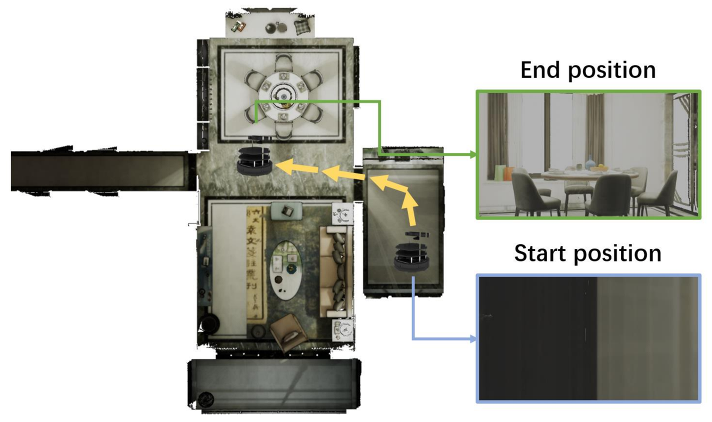
*Equal contribution; ordered alphabetically | †Corresponding authors

ECCV TEL AVIV 2022

Introduction

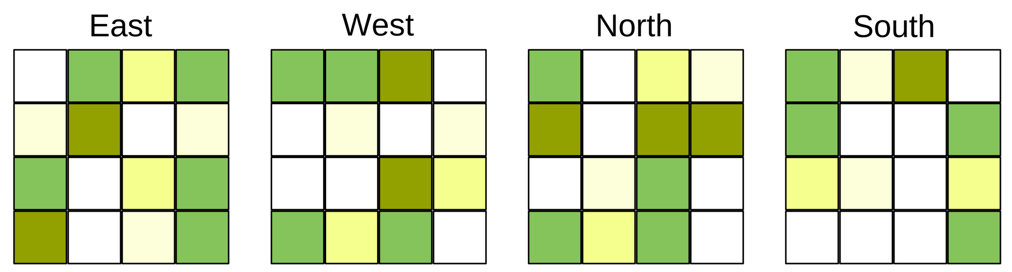
➢ Problem Definition

- **Camera localization**: Estimate the accurate **camera pose** in a **known environment.**
- **Active camera localization**: Allow the agent to **move to a new position** where the camera can be accurately localized.



➢ Drawbacks of Existing Methods

- **Camera localization in the coarse-scale discrete pose space.** Memory & computation inefficient, and not scalable to large environments and continuous camera pose space.



Grid-based Belief Map

- **Agnostic to localization-driven scene uncertainty.** Without considering the localization-driven scene uncertainty information, which is an important guidance for camera movements.

Large scene uncertainty | Small scene uncertainty

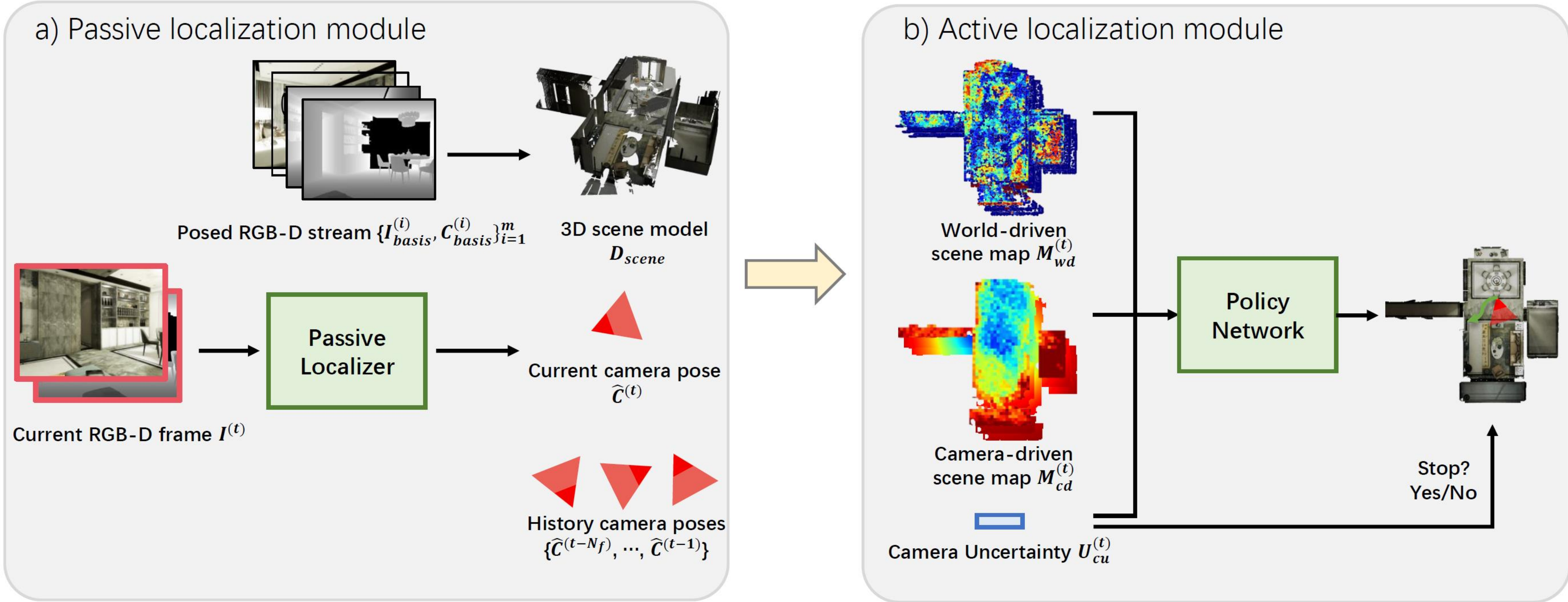
➢ Critical Questions

- **How to locate**: How to accurately localize the camera
- **Where to go**: Where it should move for accurate active localization
- **When to stop**: When it should decide to stop the camera movement

➢ Our Work

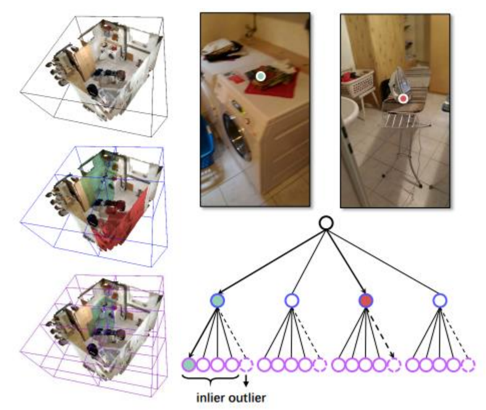
- A novel active camera localization algorithm solved by **reinforcement learning**
- Camera localization in **continuous space**
- Explicitly models the **scene uncertainty** to guide the camera movement towards localizable regions
- Explicitly models the **camera uncertainty** to determine the adaptive stop condition

Method Overview



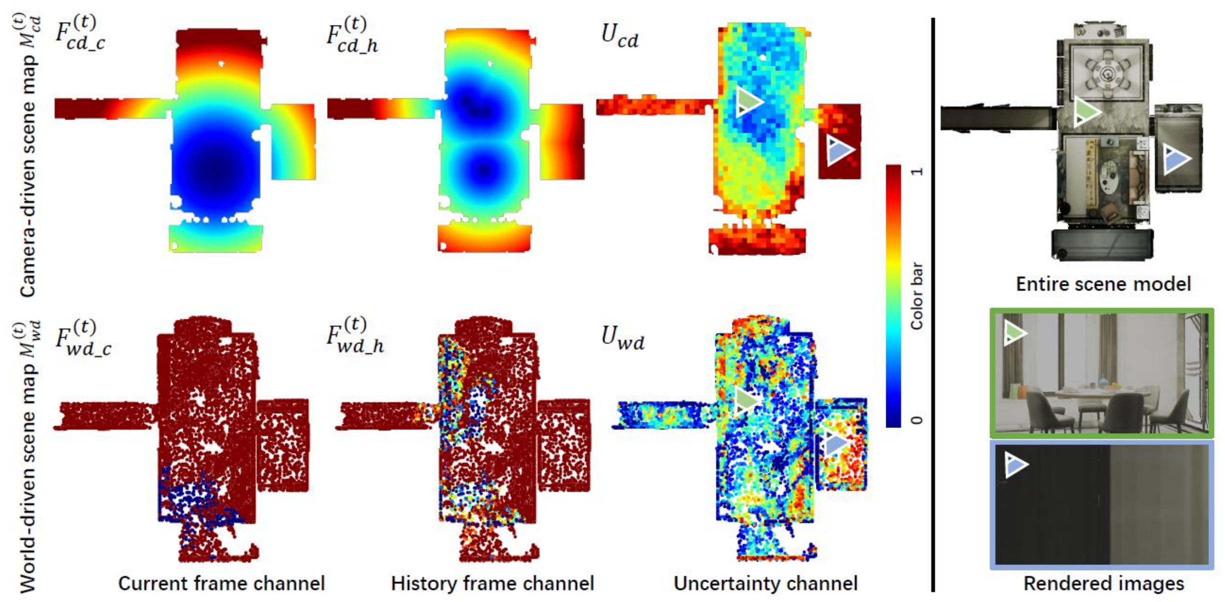
➢ Passive Localizer – How to locate

We adopt the state-of-the-art approach, **decision tree**, to achieve pose estimation in continuous space.



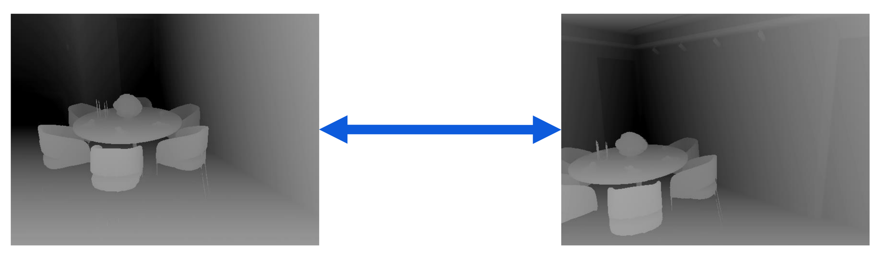
➢ Scene Uncertainty – Where to go

We describe scene uncertainty from two perspectives, **where the camera is located** and **what underlying part of the scene is observed are more effective for accurate localization.**

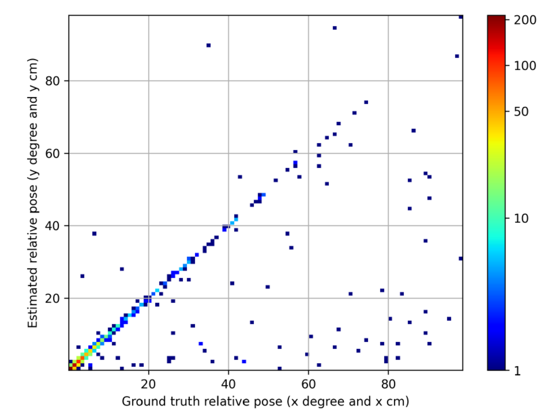


➢ Camera Uncertainty – When to stop

Prediction error is estimated by **ICP between the observed depth and the rendered depth with the predicted pose**



Observed depth | Relative pose by ICP | Rendered depth via D_{scene} and $\hat{C}^{(t)}$



Estimated relative pose VS. GT relative pose

➢ Rewards

- **Slack reward:** $R_s = -1$, punish unnecessary steps
- **Exploration reward:** $R_e = 0.1/v$, award for visiting the unseen cells where v is the visit count in the currently occupied cell

Full Pipeline

Algorithm 1 The full pipeline of our algorithm

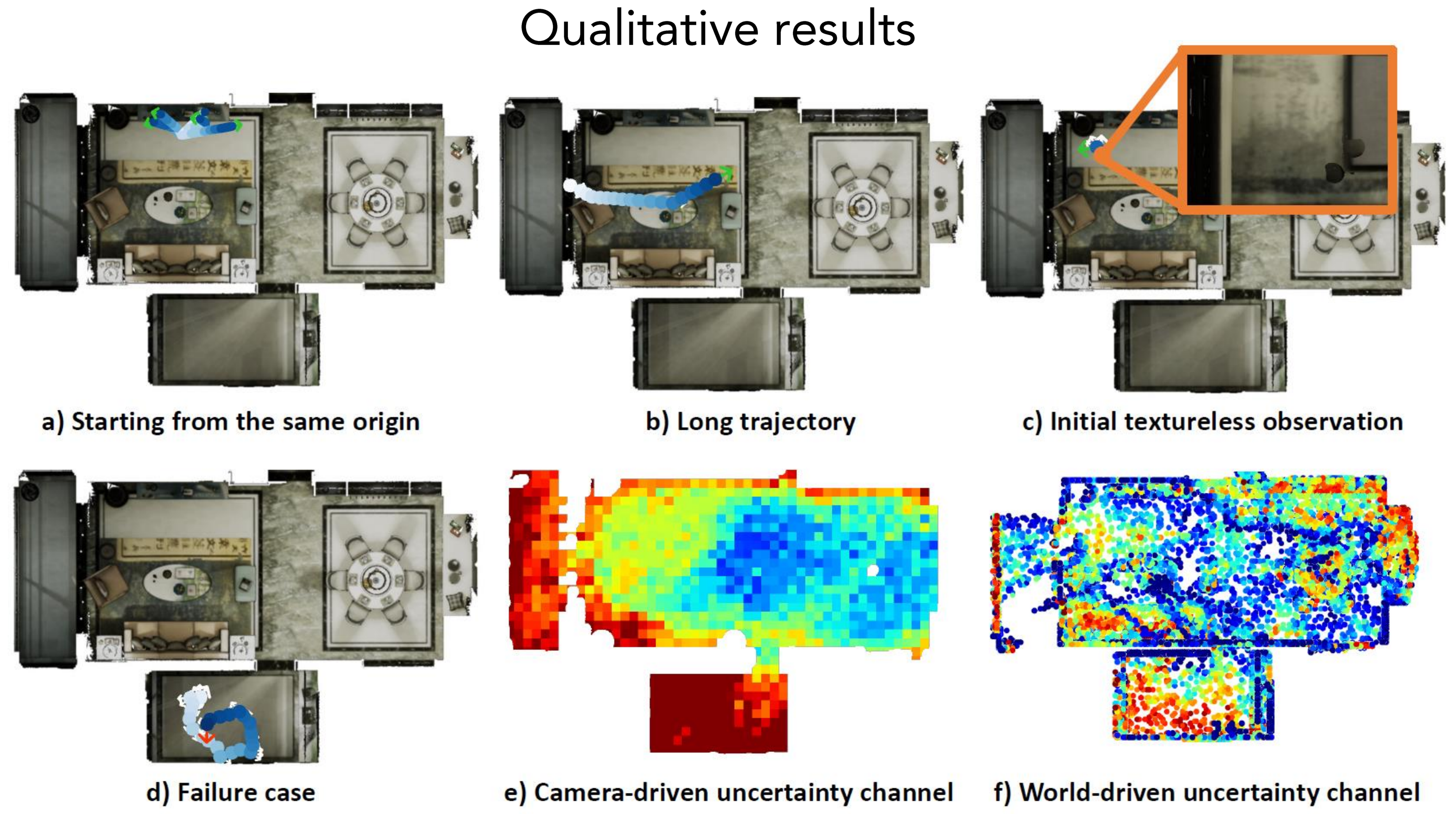
```
function PASSIVE LOC. MODULE(observation I^(t), posed RGB-D stream {I_basis^(i), C_basis^(i)}_{i=1}^m)
    if initialization then
        initialization ← false
        Adapt the passive localizer by posed RGB-D stream {I_basis^(i), C_basis^(i)}_{i=1}^m
        Construct the scene model D_scene by fusing posed RGB-D stream {I_basis^(i), C_basis^(i)}_{i=1}^m
    Current pose estimation Ĉ^(t) ← Passive localizer(I^(t))
    return Ĉ^(t)
function ACTIVE LOC. MODULE(pose estimation Ĉ^(t), scene model D_scene)
    M_wd^(t), M_cd^(t) ← Scene uncertainty computation({Ĉ^(t), D_scene})
    U_cu^(t) ← Camera uncertainty computation({Ĉ^(t), D_scene})
    Action a^(t) ← Policy network({M_wd^(t), M_cd^(t)})
    return U_cu^(t), a^(t)
procedure ENTIRE PIPELINE(posed RGB-D stream {I_basis^(i), C_basis^(i)}_{i=1}^m, accuracy threshold λ_cu)
    t ← 0
    D_scene ← NULL
    initialization ← true
    while t < maximum step length do
        Obtain the current observation I^(t)
        Ĉ^(t) ← PASSIVE LOC. MODULE(I^(t), {I_basis^(i), C_basis^(i)}_{i=1}^m)
        U_cu^(t), a^(t) ← ACTIVE LOC. MODULE(Ĉ^(t), D_scene)
        if U_cu^(t) is within λ_cu cm, λ_cu degrees then
            break
        Execute the action a^(t)
        t ← t + 1
    return Ĉ^(t)
```

Experiments

Numerical results evaluated with 5cm, 5° accuracy

	ACL-synthetic		ACL-real	
Methods	Acc (%)	#steps	Acc (%)	#steps
ANL [15]	3.25	100	3.20	100
No-movement (DecisionTree)	9.35	0	6.80	0
No-movement (DSAC)	14.90	0	7.80	0
Turn-around	25.00	12	35.20	12
Camera-descent (t+1)	61.55	22.90	61.40	26.85
Camera-descent (t+2)	55.30	22.60	59.20	25.78
Scene-descent	57.65	18.56	54.20	16.87
Ours (w/o $\mathcal{R}_e$&$M_{cd}^{(t)}$)	67.65	17.40	70.60	19.71
Ours (w/o $\mathcal{R}_e$&$M_{wd}^{(t)}$)	66.40	16.27	67.40	18.63
Ours (w/o $\mathcal{R}_e$)	72.50	18.57	73.00	20.72
Ours	**83.05**	17.33	**82.40**	17.90

Qualitative results



a) Starting from the same origin | b) Long trajectory | c) Initial textureless observation

d) Failure case | e) Camera-driven uncertainty channel | f) World-driven uncertainty channel