Learning to Look Around: Intelligently Exploring Unseen Environments for Unknown Tasks

Presenter: Jeff Hu

Sep 1st, 2022

Background & Motivation

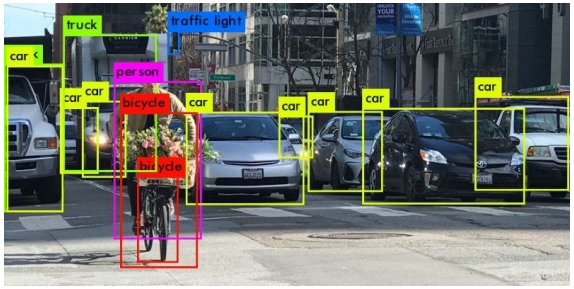
Computer Vision

- “Enables computers and systems to derive meaningful information from digital images” – From IBM Topics

Background & Motivation

Computer Vision

- “Enables computers and systems to derive meaningful information from digital images” – From IBM Topics
- Applications we have already seen:



Background & Motivation

What if we can actively determine where we want to look next?

Background & Motivation

What if we can actively determine where we want to look next?

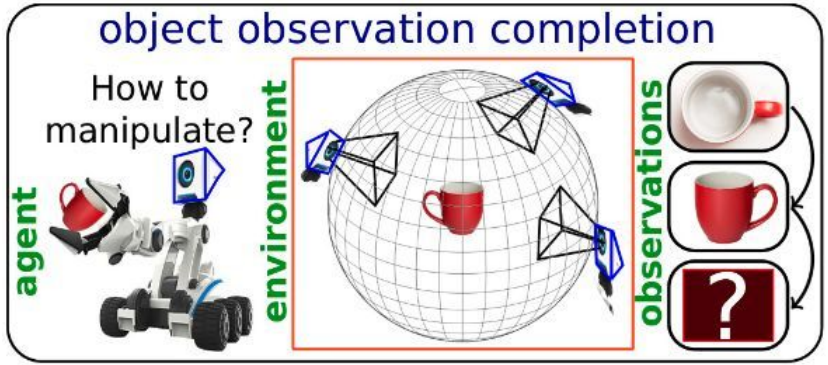
But… what does it really mean?
What is the problem we are trying to solve here?

Problem Formulation

The paper does not provide a generalized problem definition

Example: 3D object understanding task

- 3D Object
  - M*N discrete observation locations
- Camera
  - Action: 5 elevations × 5 azimuths
- Number of Observations
  - T = 4
- Goal
  - Shape reconstruction

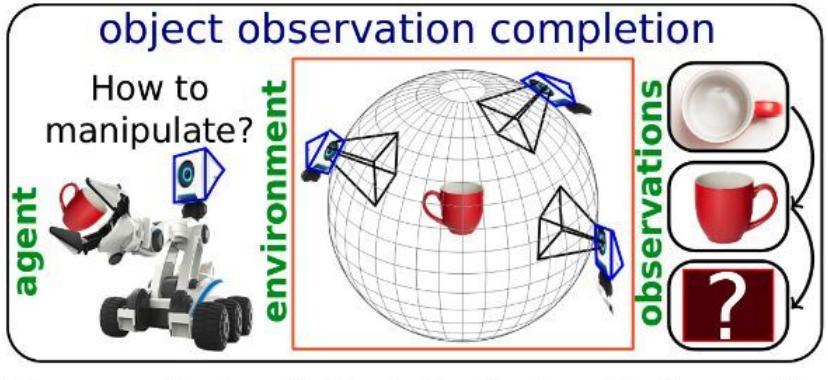


Problem Formulation

The paper does not provide a generalized problem definition

Example: 3D object understanding task

- 3D Object (Environment):
  - M*N discrete observation locations
- Camera (Active Agent):
  - Action: 5 elevations × 5 azimuths
- Number of Observations (Time):
  - T = 4
- Goal (Exploration Objective):
  - Shape reconstruction



Prior Work & limitations

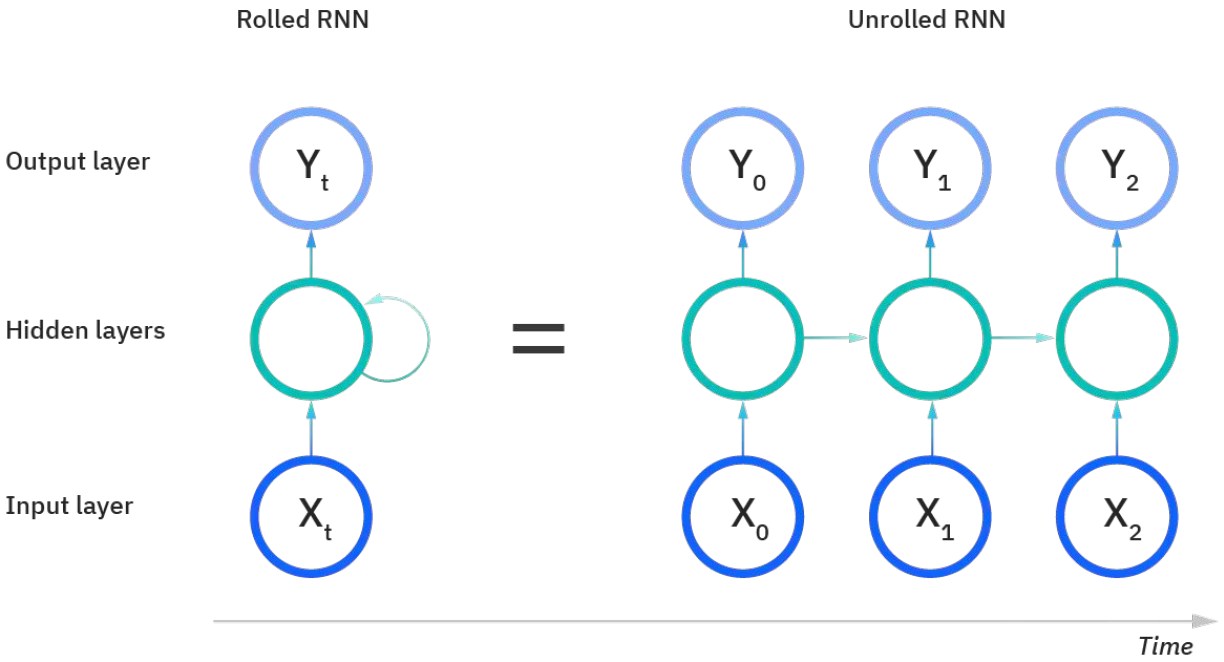
- **Saliency and attention**: find most salient regions of already captured image/video data; predict the gaze of human observer. Access to observation of the entire environment → to look for a new observation.
- **Optimal sensor placement**: how to place sensors so that they provide maximum coverage. Sensors are static → Active completion, reacting to past observations.
- **Active perception**: active object localization, action detection in video, object recognition. Pre-defined recognition tasks → general data acquisition strategy in perception; manually labeled data → unlabeled observations.
- **Active visual localization and mapping**: to limit samples needed to densely reconstruct a 3D environment geometrically. Purely geometric methods require dense observations → infer missing content with semantic and contextual clues.
- **Learning to reconstruct**: one-short reconstruction. Single view → sequence of views; image feature learning → learn action policies.

Prior Work: active object recognition

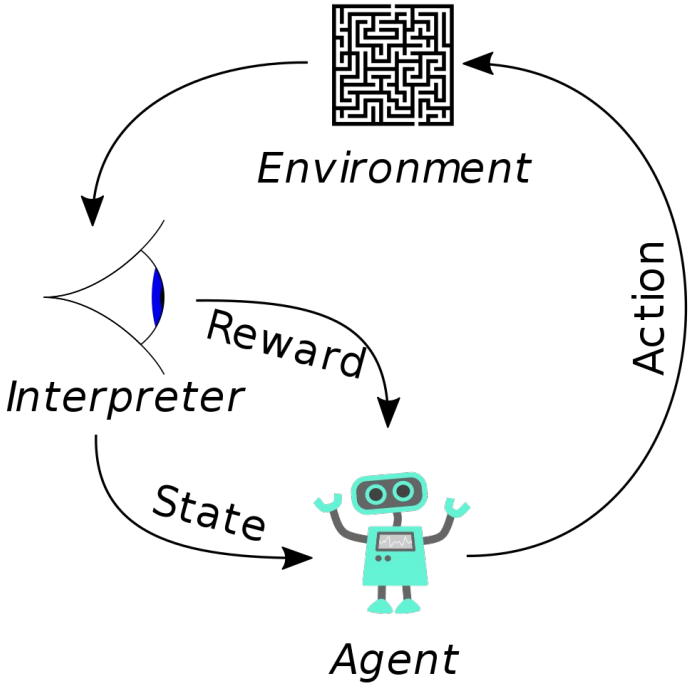
Look-ahead before you leap: end-to-end active recognition by forecasting the effect of motion, ECCV (2016)

- “Supervised”: need object label
- “Unsupervised”: we are trying to reconstruct the object, no need for label

Preliminary: RNN (LSTM)



Preliminary: Reinforcement Learning (REINFORCE)



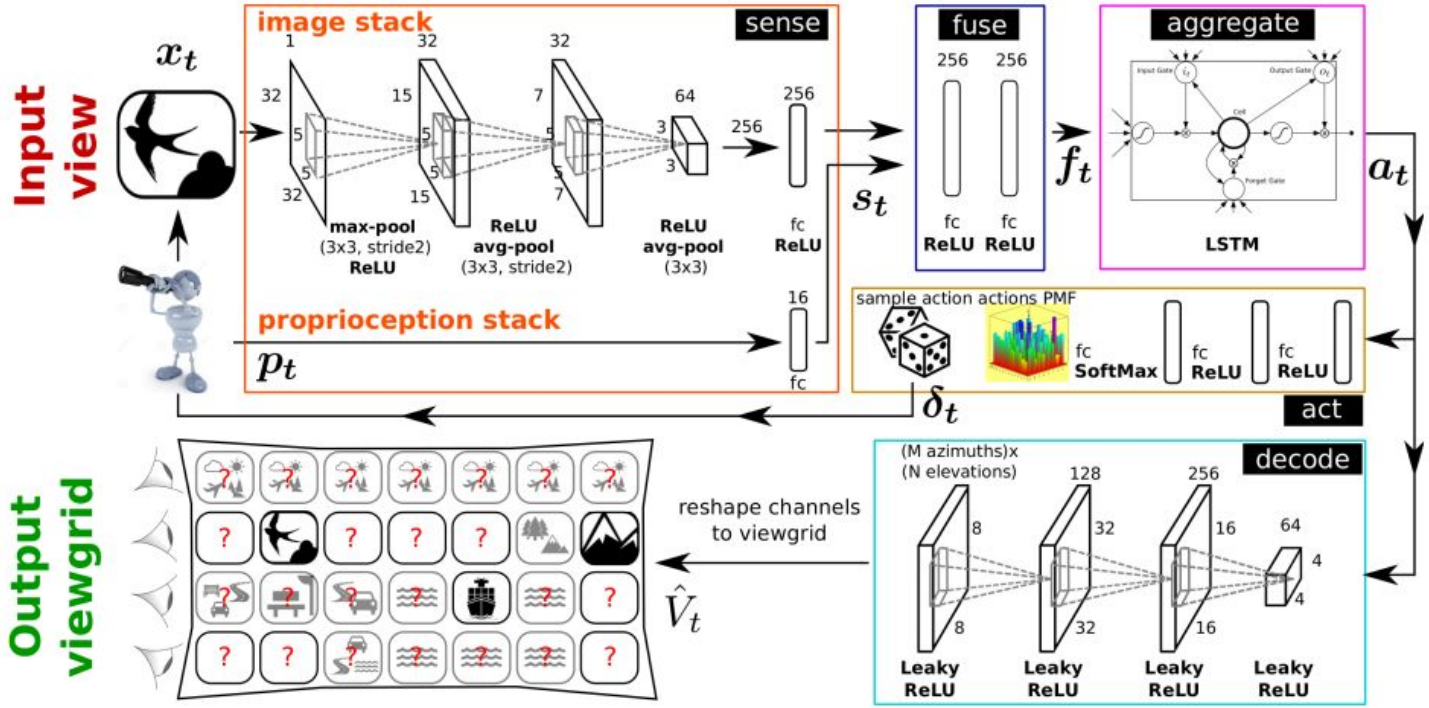
$$Policy\ gradient: E_{\pi}[\nabla_{\theta}(log\pi(s,a,\theta))R(\tau)]$$

Policy function — Score function

$$Update\ rule: \Delta\theta = \alpha * \nabla_{\theta}(log\pi(s,a,\theta))R(\tau)$$

Change in parameters — Learning rate

Full Pipeline



Loss function

$$L_T(X) = \sum_{i=1}^{MN} d(\hat{x}_T(X, \theta_i + \Delta_0), x(X, \theta_i))$$

- Specifically,

  $\nabla L_T(X)$ backpropagated via the DECODE, AGGREGATE, FUSE, SENSE modules
- ACT is stochastic as it involves sampling → use REINFORCE to handle this:

  $R(X) = -L_T(X)$

Loss function

$$L_T(X) = \sum_{i=1}^{MN} d(\hat{x}_T(X, \theta_i + \Delta_0), x(X, \theta_i))$$

- Specifically,

  $\nabla L_T(X)$ backpropagated via the DECODE, AGGREGATE, FUSE, SENSE modules
- ACT is stochastic as it involves sampling → use REINFORCE to handle this:

  $R(X) = -L_T(X)$
  - Is this actually the reason?

Tricks

- In practice, it is beneficial to penalize errors in the predicted viewgrid at every timestep rather than just at t = T:

$$L(X) = \sum_{t=1}^{T} \sum_{i=1}^{MN} d(\hat{\boldsymbol{x}}_{\boldsymbol{t}}(X, \boldsymbol{\theta}_i + \Delta_0), \boldsymbol{x}(X, \boldsymbol{\theta}_i))$$

- Pretrain the entire network with T = 1
  - Essentially, no action involved
  - We will talk more about this later

Experimental Setup: Active observation completion

Datasets:

SUN360 (scene)	ModelNet (object)
• 26 category • 32x32 views from 5 camera elevations and 8 azimuths • per-timestep motions within 3x5 • Training episode length T = 6	• Train on seen (ModelNet-40 \ ModelNet-10); unseen (ModelNet-10) • 32x32 views from 7 camera elevations and 12 azimuths • per-timestep motions within 5x5 • Training episode length T = 4

Baselines: *ours* compared with

- *1-view*: the method trained with T=1
- *random*: the method with randomly action selection module
- *large-action*: largest allowable action
- *peek-saliency*: most salient view within reach at each timestep

Harel, Jonathan, Christof Koch, and Pietro Perona. "Graph-based visual saliency." Advances in neural information processing systems 19 (2006).

Result: Active observation completion

Dataset→	SUN360		ModelNet (seen classes)		ModelNet (unseen classes)	
Method↓ — Metric→	MSE(x1000)	Improvement	MSE(x1000)	Improvement	MSE(x1000)	Improvement
1-view	39.40	-	3.83	-	7.38	-
random	31.88	19.09%	3.46	9.66%	6.22	15.72%
large-action	30.76	21.93%	3.44	10.18%	6.16	16.53%
peek-saliency [23]	27.00	31.47%	3.47	9.40%	6.35	13.96%
ours	**23.16**	**41.22%**	**3.25**	**15.14%**	**5.65**	**23.44%**

- Improvements larger on more difficult datasets (SUN360 > unseen ModelNet > seen ModelNet)

[23] Harel, Jonathan, Christof Koch, and Pietro Perona. "Graph-based visual saliency." Advances in neural information processing systems 19 (2006).

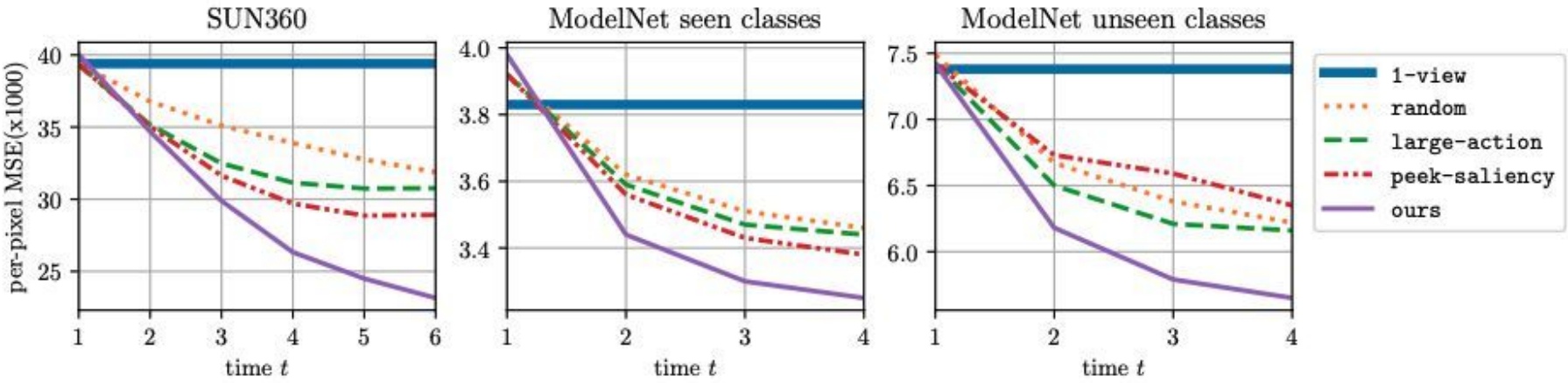
Result: Active observation completion

Dataset→	SUN360		ModelNet (seen classes)		ModelNet (unseen classes)	
Method↓ — Metric→	MSE(x1000)	Improvement	MSE(x1000)	Improvement	MSE(x1000)	Improvement
1-view	39.40	-	3.83	-	7.38	-
random	31.88	19.09%	3.46	9.66%	6.22	15.72%
large-action	30.76	21.93%	3.44	10.18%	6.16	16.53%
peek-saliency [23]	27.00	31.47%	3.47	9.40%	6.35	13.96%
ours	**23.16**	**41.22%**	**3.25**	**15.14%**	**5.65**	**23.44%**

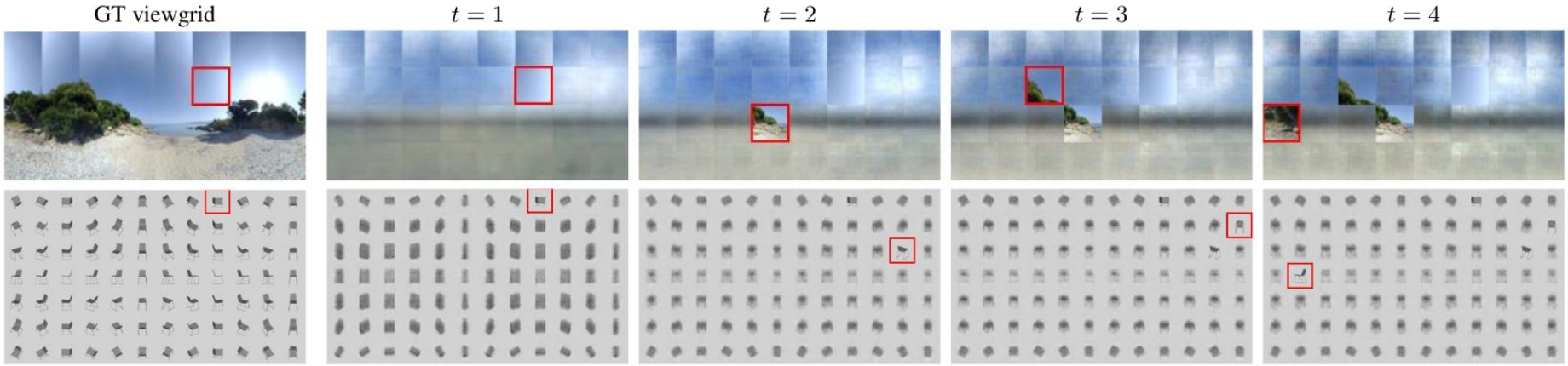
- Improvements larger on more difficult datasets (SUN360 > unseen ModelNet > seen ModelNet)
- These baselines are all relatively weak

[23] Harel, Jonathan, Christof Koch, and Pietro Perona. "Graph-based visual saliency." Advances in neural information processing systems 19 (2006).

Result: Active observation completion



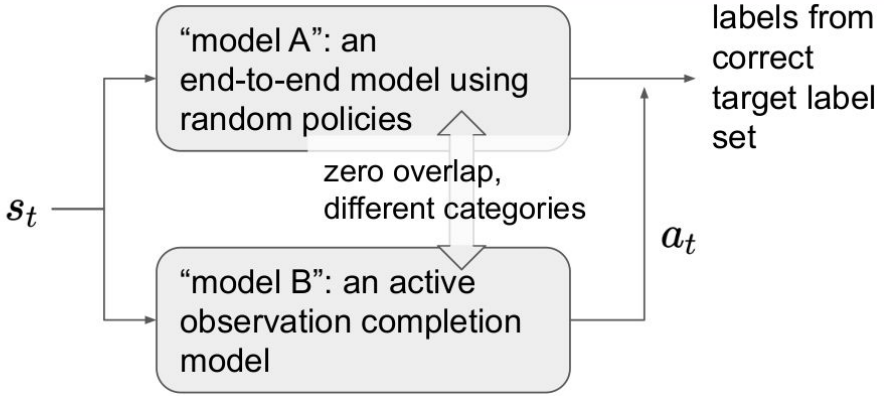
Result Visualization: Active observation completion



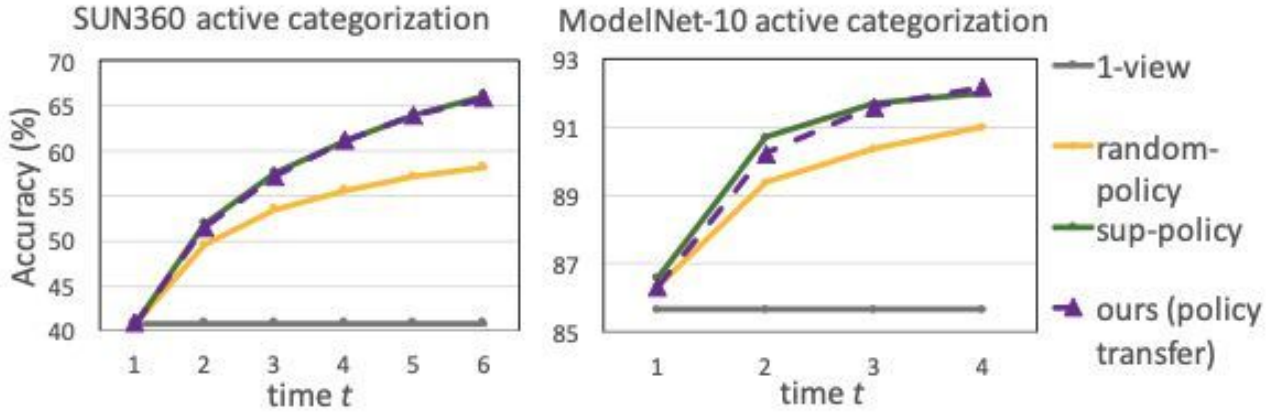
Policy Transfer

Objective: to inject the generic look-around policy into unseen tasks in unseen environments.

Approach:



Result: Policy Transfer



ours outperforms 1-view and random-policy, on par with sup-policy

Limitations & Critiques

Limitations & Critiques

- Unclear why co-training the policy and the reconstruction network
  - Essentially 2 agents cooperating with each other
    - Consider: the policy learns to stay fixed; the reconstruction network learns to reconstruct only using the first observation
      - This is a Nash equilibrium!
  - Policy training relies on a good reconstruction network (trick 2)

Tricks

- In practice, it is beneficial to penalize errors in the predicted viewgrid at every timestep rather than just at t = T:

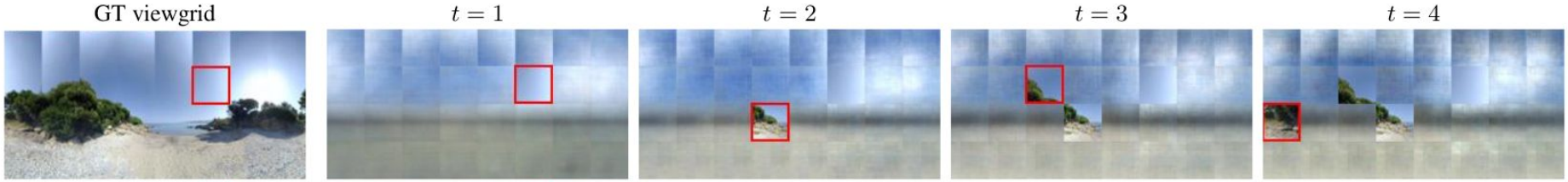
$$L(X) = \sum_{t=1}^{T} \sum_{i=1}^{MN} d(\hat{\boldsymbol{x}}_{\boldsymbol{t}}(X, \boldsymbol{\theta}_i + \Delta_0), \boldsymbol{x}(X, \boldsymbol{\theta}_i))$$

- Pretrain the entire network with T = 1
  - Essentially, no action involved
  - This is essentially making sure that at the start of the training the reconstruction network is already reasonable

Limitations & Critiques

- Unclear why co-training the policy and the reconstruction network
  - Essentially 2 agents cooperating with each other
    - Consider: the policy learns to stay fixed; the reconstruction network learns to reconstruct only using the first observation
      - This is a Nash equilibrium!
  - Policy training relies on a good reconstruction network (trick 2)
- How well can the policy generalize?
  - What if the actor need to complete certain physical tasks while observing?
    - e.g.: mobile manipulator
  - What about if the total time T changes?

Result Visualization (limitations)



What if T changes?

Things to be improved…

- “ Exploration episodes are shown at https://goo.gl/BgWX3W”

- No code
- Not enough visual proof & no theoretical proof

Future works

Train faster and converge to better policies?

- Sidekick policy learning [Ramakrishnan et al. 2018]

Geometry awareness (cross-object occlusion)?

- Geometry-aware RNN [Cheng et al. 2018]

Extended Readings

- Yang, J., Ren, Z., Xu, M., Chen, X., Crandall, D., Parikh, D., & Batra, D. (2019). Embodied amodal recognition: Learning to move to perceive objects. Proceedings of the IEEE International Conference on Computer Vision, 2019-October, 2040–2050. https://doi.org/10.1109/ICCV.2019.00213
- Ramakrishnan, S. K., Al-Halah, Z., & Grauman, K. (2020). Occupancy Anticipation for Efficient Exploration and Navigation. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 12350 LNCS, 400–418. https://doi.org/10.1007/978-3-030-58558-7_24
- Ramakrishnan, S. K., Jayaraman, D., & Grauman, K. (2021). An Exploration of Embodied Visual Exploration. International Journal of Computer Vision, 129(5), 1616–1649. https://doi.org/10.1007/s11263-021-01437-z
- Cheng, R., Wang, Z., & Fragkiadaki, K. (2018). Geometry-aware recurrent neural networks for active visual recognition. Advances in Neural Information Processing Systems, 2018-Decem(Nips), 5081–5091. https://arxiv.org/pdf/1811.01292.pdf
- Ramakrishnan, S. K., & Grauman, K. (2018). Sidekick policy learning for active visual exploration. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 11216 LNCS, 424–442. https://doi.org/10.1007/978-3-030-01258-8_26

Summary

- **Problem**: How can a visual agent autonomously capture good observations?
- **Why important?** Crucial step towards embodied, active agents in novel environments
- **Key limitations**: limited generalizability, weak theoretical analysis
- **Advantages**: transferability, “unsupervised” training
- **Key insights**: the agent is rewarded for actions that reduce its uncertainty about the unobserved portions of the environment
- **What did they demonstrate by this insight?**
  - SOTA performance on active observation completion tasks
  - First to accomplish “policy transfer” between tasks

Thank you!