A Visual Navigation Perspective for Category-Level Object Pose Estimation

Jiaxin Guo[1,2], Fangxun Zhong[2], Rong Xiong[1], Yunhui Liu[2], Yue Wang[1,*], and Yiyi Liao[1,**]

[1] Zhejiang University, Hangzhou, China

[2] The Chinese University of Hong Kong, Hong Kong, China

1. Introduction

❖ **Task:** Category-level object pose estimation based on a single monocular image.

❖ **Analysis-by-synthesis:** To sequentially update a set of latent variables of the generative model until the generated image best agrees with the observation.

❖ **Problem: Convergence and efficiency** are two challenges based on gradient descent (GD).

❖ **Goal:** View the inference as a **visual navigation task** and **investigate what is a good navigation policy** for this specific task.

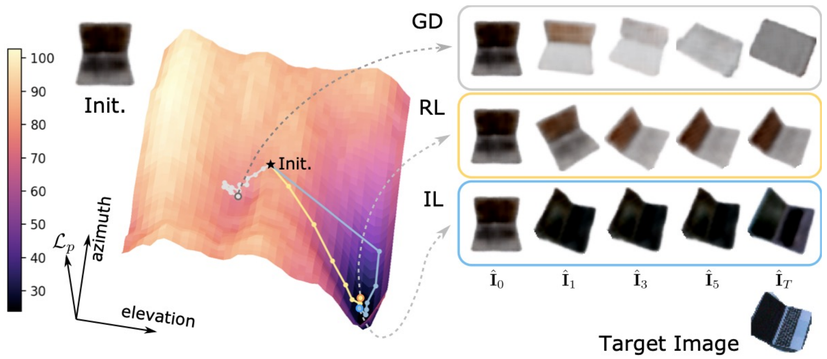


Fig. 1: Inference of Analysis-by-Synthesis. Color bar: perceptual loss; GD/RL/IL: Navigation trajectories; $\hat{I}_0$: The same initialization synthesized image;

2. Problem Formulation

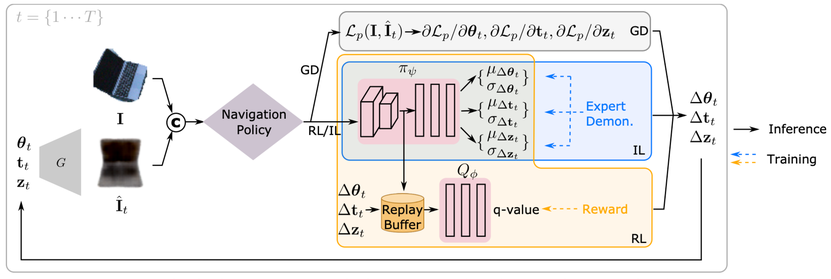


Fig. 2: Category-Level Object Pose Estimation as Visual Navigation.

❖ Model the pose estimation as a Markov decision process (MDP).

❖ Input **synthesized image** $\hat{\mathbf{I}}_t = G(\boldsymbol{\theta}_t, \boldsymbol{t}_t, \boldsymbol{z}_t)$ and **target image** $\mathbf{I}$ to update iteratively taking T steps of actions :

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \Delta\boldsymbol{\theta}_t,\ \mathbf{t}_{t+1} = \mathbf{t}_t + \Delta\mathbf{t}_t,\ \mathbf{z}_{t+1} = \mathbf{z}_t + \Delta\mathbf{z}_t$$

3. Visual Navigation Policy

❖ **Gradient Descent:** When G is differentiable, update using GD straightforwardly:

$$\pi(\mathbf{a}_t|\mathbf{o}_t) = -\lambda\frac{\partial\mathcal{L}_p}{\partial\boldsymbol{\theta}_t}, -\lambda\frac{\partial\mathcal{L}_p}{\partial\mathbf{t}_t}, -\lambda\frac{\partial\mathcal{L}_p}{\partial\mathbf{z}_t}$$

❖ **Reinforcement Learning:** Recover both the object appearance and its pose simultaneously by maximum the reward:

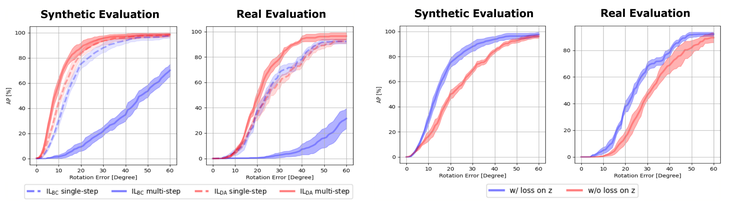
$$r_t = -\lambda_1\|\mathbf{q}(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t) - \mathbf{q}(\Delta\boldsymbol{\theta}_t)\|_2^2 - \lambda_2\|(\mathbf{t}^* - \mathbf{t}_t) - \Delta\mathbf{t}_t\|_2^2 - \lambda_3\|(\mathbf{z}^* - \mathbf{z}_t) - \Delta\mathbf{z}_t\|_2^2$$

❖ **Imitation Learning:** Same network structure as RL, train Behavior Cloning (BC) and Dataset Aggregation (DAgger) via the loss:

$$\mathcal{L}_{IL} = \lambda_1\|\mathbf{q}(\boldsymbol{\theta}^* - \boldsymbol{\theta}_t) - \mathbf{q}(\Delta\boldsymbol{\theta}_t)\|_2^2 + \lambda_2\|(\mathbf{t}^* - \mathbf{t}_t) - \Delta\mathbf{t}_t\|_2^2 + \lambda_3\|(\mathbf{z}^* - \mathbf{z}) - \Delta\mathbf{z}\|_2^2$$

4. Policy Analysis

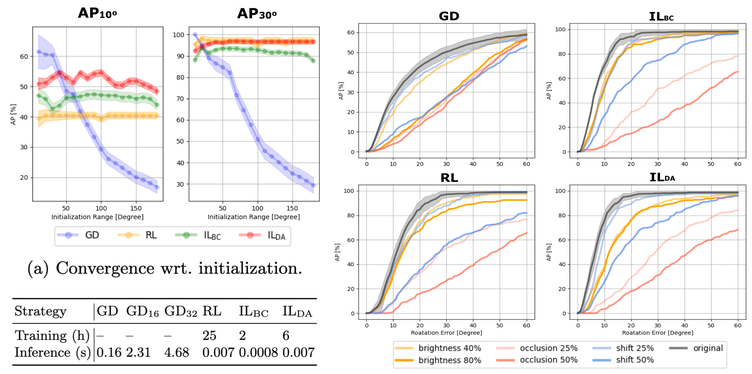
❖ **How are Policies Affected by Design Choices?**



(a) Multi-Step v.s. Single-Step. (b) Prediction of Latent Code.

Fig. 3: Effect of Design Choices on synthetic & real-world images.

❖ **What is a Good Navigation Policy?**



(a) Convergence wrt. initialization.

Strategy	GD	GD_{16}	GD_{32}	RL	IL_{BC}	IL_{DA}
Training (h)	–	–	–	25	2	6
Inference (s)	0.16	2.31	4.68	0.007	0.0008	0.007

(b) Efficiency. (c) Robustness.

Fig. 4: Convergence, Robustness, and Efficiency of different navigation policies.

❖ **RL/IL:** better convergence and efficiency given arbitrary initialization.

❖ **GD:** better precision and robustness given a good initialization.

❖ **Simple hybrid solution:** RL/IL with GD for fine-tuning.

5. Experimental Results

❖ RL and IL_{DA} achieve competitive performance compared to Chen et al. but are **remarkably more efficient**.

❖ The simple hybrid approaches, **RL w/ GD** and **IL_{DA} w/ GD**, often lead to better performance.

Dataset	Metric	NOCS* [50]	VGG [48]	iNeRF [54]	Chen [8]	GD	RL	IL_{DA}	RL w/ GD	IL_{DA} w/ GD
REAL275 Dataset (Symmetry)	$AP_{10°}$	32.8	6.4	21.0	24.0	20.5	18.6	21.6	24.8	**25.0**
	$AP_{30°}$	66.5	34.8	88.7	92.1	86.2	91.7	92.5	93.6	**94.2**
	$AP_{60°}$	99.3	76.3	97.1	**99.9**	96.7	98.8	99.6	99.6	**99.9**
	AP_{5cm}	93.4	7.8	11.9	12.7	11.9	11.8	12.4	13.2	**14.6**
	AP_{10cm}	95.0	23.7	26.1	27.4	24.5	23.9	**29.1**	27.2	28.8
	AP_{15cm}	97.3	38.1	43.8	**46.9**	41.4	39.5	42.3	42.6	46.4
REAL275 Dataset (Asymmetry)	$AP_{10°}$	20.5	0.6	5.1	**6.9**	5.0	5.1	4.8	6.5	6.8
	$AP_{30°}$	55.5	12.4	43.1	59.5	21.1	51.6	53.5	58.7	**60.0**
	$AP_{60°}$	93.3	35.1	62.8	79.2	35.0	74.5	80.6	76.3	**82.3**
	AP_{5cm}	87.7	10.3	7.7	12.1	9.8	9.7	**17.5**	12.8	12.5
	AP_{10cm}	98.2	38.1	33.2	42.4	27.7	41.7	**52.2**	42.2	46.8
	AP_{15cm}	99.5	61.8	48.7	73.1	50.6	71.6	**75.8**	73.0	72.8
Cars Dataset	$AP_{10°}$	\	5.6	31.7	51.8	21.3	42.4	47.6	62.9	**65.3**
	$AP_{30°}$	\	15.4	45.4	85.5	33.7	92.8	93.5	93.8	**94.1**
	$AP_{60°}$	\	32.8	56.6	93.7	37.4	94.2	98.2	97.7	**98.8**
	AP_{1cm}	\	8.9	12.4	35.7	9.6	27.2	35.5	29.1	**36.7**
	AP_{3cm}	\	35.2	41.6	75.8	32.7	71.8	**76.3**	72.5	75.6
	AP_{6cm}	\	52.1	68.3	85.7	60.1	91.8	92.0	91.4	**92.9**
Faces Dataset	$AP_{5°}$	\	5.3	4.6	24.8	2.0	17.3	**25.8**	15.4	23.1
	$AP_{15°}$	\	32.8	42.6	88.7	35.9	84.2	89.5	88.4	**90.8**
	$AP_{30°}$	\	71.1	80.9	92.5	81.2	98.6	98.3	**99.5**	99.1
	AP_{1cm}	\	11.0	18.2	27.4	14.3	**27.9**	25.9	26.7	25.3
	AP_{3cm}	\	41.0	59.6	**92.6**	53.8	86.3	90.1	87.8	91.5
	AP_{6cm}	\	72.9	85.7	98.5	82.3	97.2	97.7	98.5	**99.5**
Mean	AP_{rot}	\	27.4	48.3	66.6	39.7	64.2	67.2	68.0	**70.0**
	AP_{tran}	\	33.4	38.1	52.5	34.9	50.0	**53.9**	51.4	53.6

*NOCS is based on RGB-D while the others are based on RGB images.

TABLE 1: Quantitative Comparison of category-level pose estimation on different datasets.

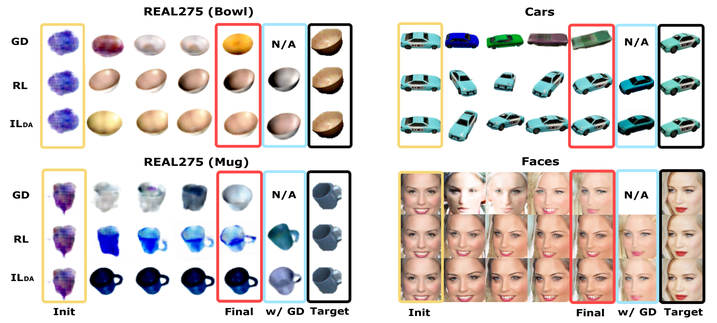


Fig. 5: Qualitative Comparison of different strategies. RL and IL_{DA} show the synthesized image after adding 10 steps of GD.

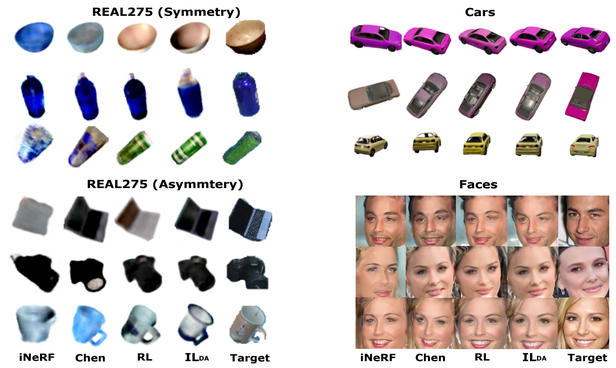


Fig. 6: Qualitative Comparisons on different datasets.