Vision-as-Inverse Graphics Agent [ECCV 2026]

Vision-as-Inverse-Graphics Agent via Interleaved Multimodal Reasoning

Shaofeng Yin[1,4*], Jiaxin Ge[1], Zora Zhiruo Wang[2], Chenyang Wang[1,4], Xiuyu Li[1]
Michael J. Black[3], Trevor Darrell[1], Angjoo Kanazawa[1], and Haiwen Feng[1,3,4†]

[1]University of California, Berkeley [2]Carnegie Mellon University
[3]Max Planck Institute for Intelligent Systems [4]Impossible, Inc.

[*]shaofeng.yin@impossible.place [†]haiwen.feng@berkeley.edu



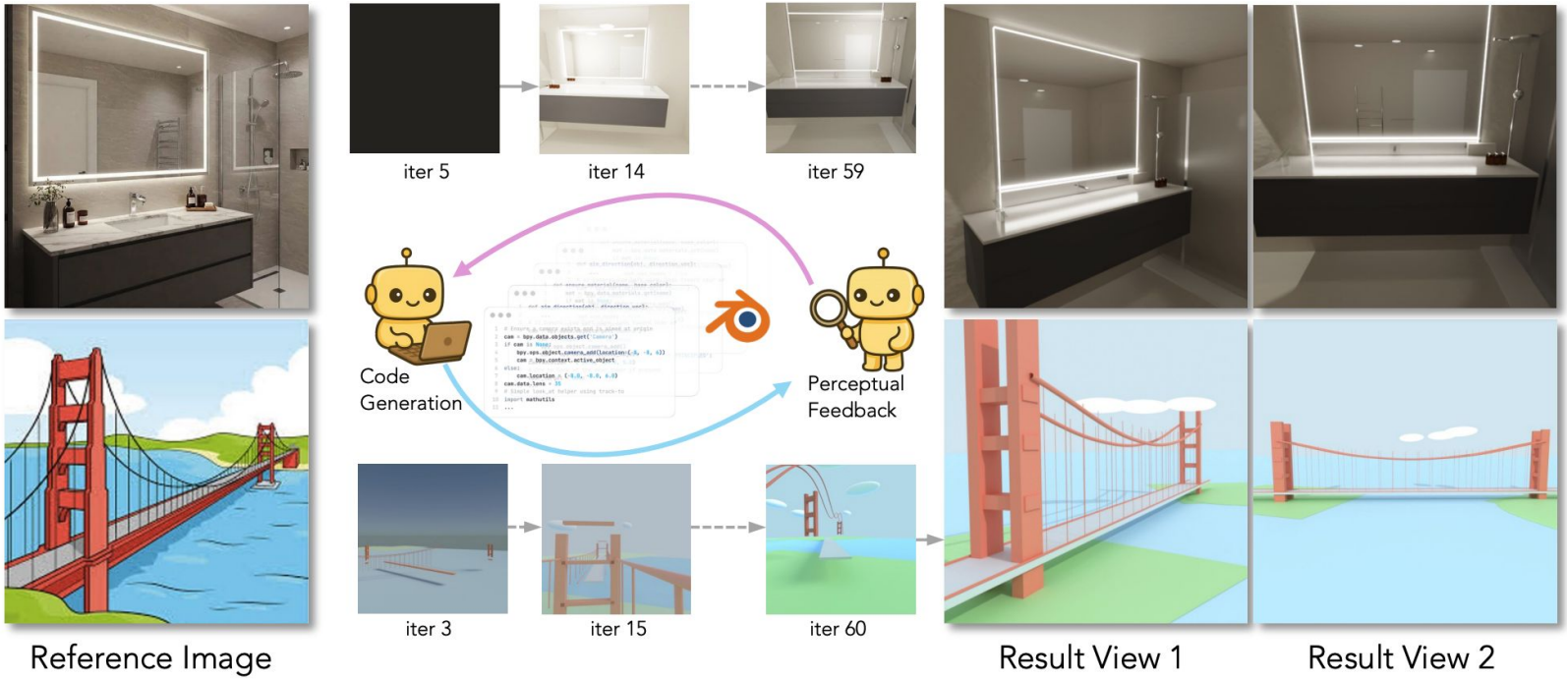
fugtemypt123.github.io/VIGA-website

introduction

usually, a 3D scene → image # rendering

in this paper, image → inferred 3D scene

conventionally, 3D scene reconstruction is done directly, VIGA recovers executable graphics program that constructs the scene



introduction

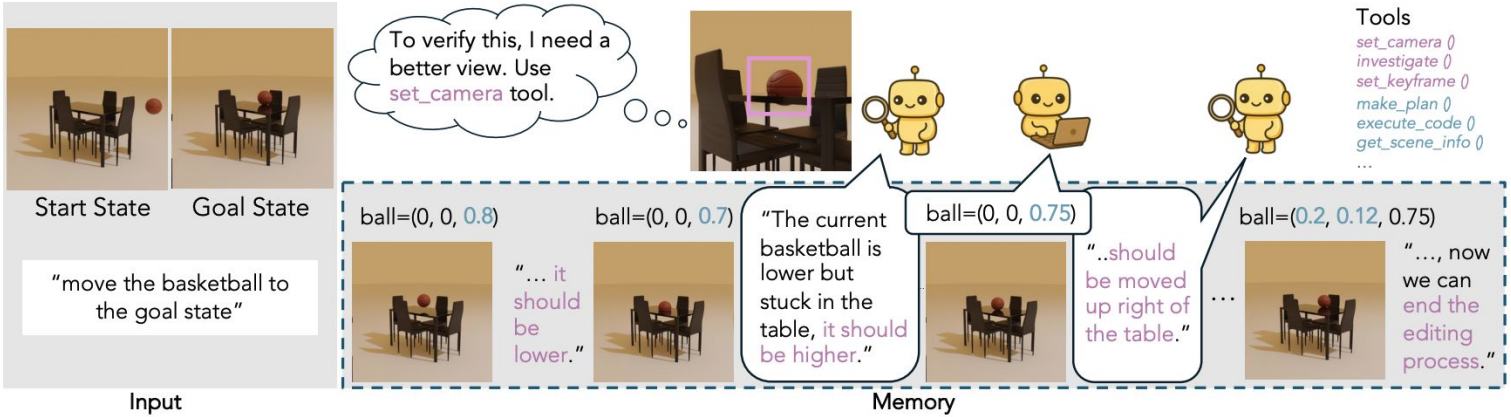
VIGA is a closed-loop iterative process where programs are realized in Blender

I ← image of the scene

while true:

a. Vlm agent plans and generates code
b. Blender executes that code and renders the scene (pixels)
c. Vlm agent inspects the generated scene & generates updated code to fix issues
d. Repeat

VIGA is training-free and lets the VLM use whatever tools it needs to generate the blender programs



VIGA algorithm

Require: Target specification s_g^*

Initialize contextual memory $M_0 \leftarrow \emptyset$
for $t = 0, 1, \ldots, T-1$ **do**
    // Phase 1: Symbolic Reasoning from Visual Input
    $\tau_t^{plan} \leftarrow \mathcal{A}(s_g^*, M_t)$ ▷ Formulates high-level plan
    if τ_t^{plan} includes **end_process then break**
    end if
    $p_t \leftarrow \mathcal{A}(s_g^*, M_t, \tau_t^{plan})$ ▷ Synthesizes executable program
    // Phase 2: Cross-Modal Execution
    $s_t \leftarrow \text{exec}(p_t)$ ▷ Renders concrete geometric states
    // Phase 3: Visual-to-Symbolic Feedback
    $\tau_t^{obs} \leftarrow \mathcal{A}(s_g^*, s_t, M_t)$ ▷ Explores the rendered scene
    $f_t \leftarrow \mathcal{A}(s_g^*, s_t, \tau_t^{obs}, M_t)$ ▷ Translates into actionable feedback
    // Phase 4: Context Memory Update (Sec. 3.2)
    $M_{t+1} \leftarrow \text{Tail}_L(M_t \cup \{p_t, f_t\})$
end for

M_t -- memory of previous edits and feedback

Termination is also VLM dependent -- when VLM outputs end process

Best-of-4 generation at each iteration using CLIP score

semantic skill library

to make blender manageable, VIGA provides a skill library to the agent

these skills are divided into observation and modification

they serve as abstractions -- agent control (end_process eg) or blender functions

execute_code() → the main function contains three things: thought, diff, code

Semantic Skill	Encapsulated APIs	Capability
State Observation		
Spatial Navigation	`set_camera`, `investigate`	Actively adjust viewpoints to inspect regions and bypass occlusions.
Temporal Inspection	`set_keyframe`	Navigate the 4D timeline to evaluate dynamic physical motions.
Structural Query	`get_scene_info`, `set_visibility`	Query geometric attributes and toggle object visibility.
Diagnostic Rendering	`initialize_view`	Render canonical views and compute target bounding boxes.
State Modification		
Asset Instantiation	`init_scene`, `get_better_assets`	Retrieve or generate high-fidelity 3D assets via external modules to populate scenes.
State Execution	`execute_code`	Apply synthesized programs to persistently update scene states.
Trajectory Control	`undo_action`, `end_process`	Revert erroneous edits or cleanly terminate the reasoning loop.

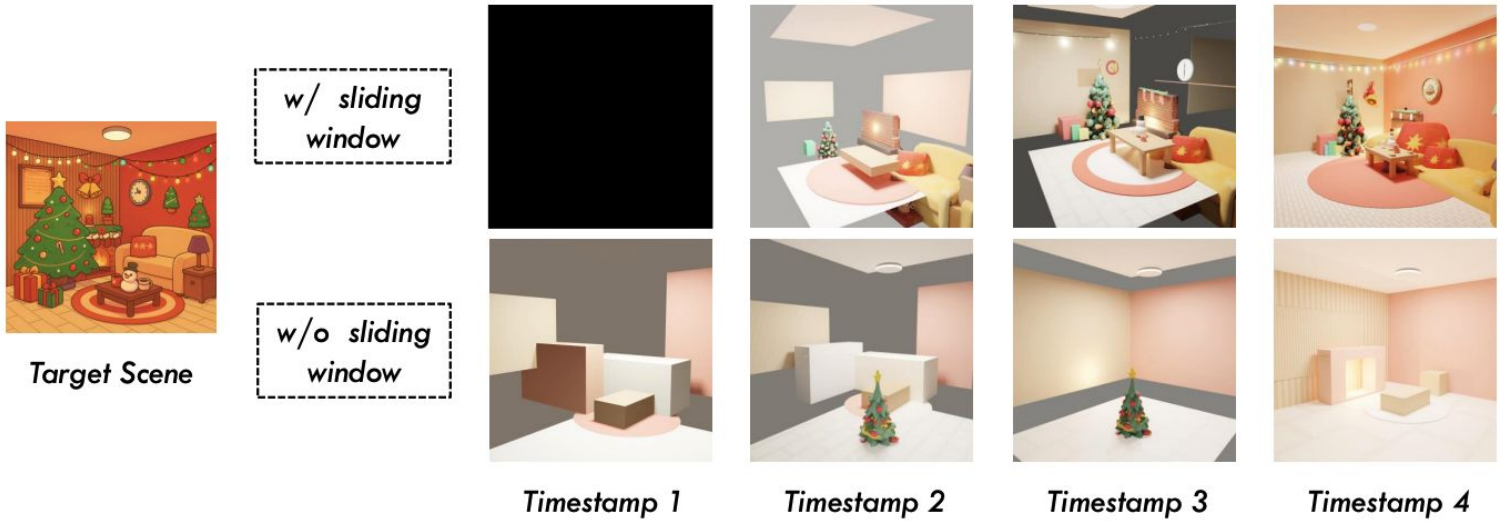
```
execute_code(
    thought="The ball
is too low.",
    code_diff="""
    -:
[ball.location.z =
0.70]
    +:
[ball.location.z =
0.75]
    """,
    code="...
complete updated
Blender script ...")
```

memory is sliding windowed

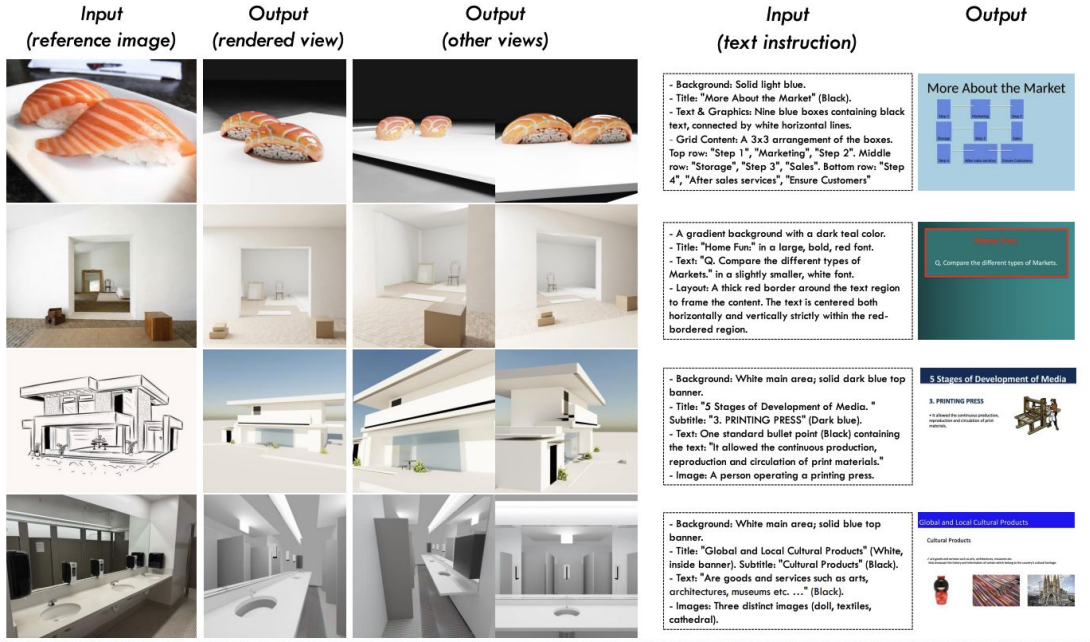
memory is a multimodal history and contains:

1. plans and natural-language feedback
2. executable programs and edits
3. rendered scene observations

a sliding window of fixed length makes sure that context bloat is mitigated



results -- qualitative



results -- quantitative

BlenderBench

Model	Setting	Task 1			Task 2			Task 3			Impr. (%)
		PL ↓	N-CLIP ↓	VLM ↑	PL ↓	N-CLIP ↓	VLM ↑	PL ↓	N-CLIP ↓	VLM ↑	
GPT-4o	One-Shot	48.16	64.17	0.58	7.36	7.12	2.75	30.14	38.69	0.25	-
	BlenderAlchemy (best-of-1)	10.62	**17.68**	**1.75**	6.13	7.71	3.08	19.60	26.10	0.67	67.56
	VIGA (best-of-1)	**8.56** (+19.40%)	18.19 (-2.89%)	1.44 (-17.71%)	**5.11** (+16.64%)	**4.33** (+43.84%)	**3.58** (+16.23%)	**14.51** (+25.97%)	**17.91** (+31.40%)	**1.53** (+128.36%)	**113.96** (+68.69%)
	BlenderAlchemy (best-of-4)	14.50	19.57	1.75	**1.95**	**2.47**	3.53	20.62	25.79	0.56	77.48
	VIGA (best-of-4)	**5.47** (+62.28%)	**6.10** (+68.84%)	**3.25** (+85.71%)	2.94 (-50.77%)	3.50 (-41.70%)	**3.83** (+8.50%)	**12.62** (+38.79%)	**22.84** (+11.43%)	**1.61** (+187.50%)	**159.19** (+105.45%)
Qwen3-VL-8B	One-Shot	60.82	78.16	0.28	33.14	37.51	1.61	22.81	22.98	1.25	-
	BlenderAlchemy (best-of-1)	25.61	37.93	0.36	10.69	13.92	2.64	30.45	23.91	0.64	27.36
	VIGA (best-of-1)	**10.54** (+58.86%)	**17.27** (+54.45%)	**1.31** (+263.89%)	**3.85** (+63.98%)	**3.34** (+76.06%)	**3.33** (+26.14%)	**11.35** (+62.75%)	**6.86** (+71.33%)	**2.25** (+251.56%)	**112.79** (+312.20%)
	BlenderAlchemy (best-of-4)	11.57	17.58	1.19	5.38	4.64	2.94	23.62	15.67	0.93	82.24
	VIGA (best-of-4)	**8.80** (+23.92%)	**15.67** (+10.88%)	**1.38** (+15.97%)	**5.02** (+6.69%)	**3.86** (+16.81%)	**3.08** (+4.76%)	**9.08** (+61.54%)	**6.94** (+55.73%)	**2.02** (+117.20%)	**112.87** (+37.20%)

PL: photometric loss (RGB diff.)

N-CLIP: negative CLIP (CLIP-space similarity -ve)

VLM: VLM Score (use another VLM to assess)

results -- quantitative

Model	Setting	Shape		Place		Geom		Light		Mat		Impr. (%)
		PL ↓	NC ↓	PL ↓	NC ↓	PL ↓	NC ↓	PL ↓	NC ↓	PL ↓	NC ↓	
GPT-4o	One-Shot	7.94	19.96	11.86	29.31	18.12	24.91	2.06	2.17	8.78	15.25	-
	B-Alchemy (bo1)	7.82	20.19	11.57	28.07	15.58	20.61	1.44	1.82	7.22	12.05	12.33
	VIGA (bo1)	**6.70** (+14.3%)	**16.70** (+17.3%)	**9.89** (+14.5%)	**26.87** (+4.3%)	**11.82** (+24.1%)	**15.76** (+23.5%)	**1.43** (+0.7%)	**1.41** (+22.5%)	**6.11** (+15.4%)	**9.39** (+22.1%)	**26.88** (+118.1%)
	B-Alchemy (bo4)	5.78	17.19	10.29	25.58	9.47	11.68	0.97	1.23	9.47	12.20	27.63
	VIGA (bo4)	**5.45** (+5.7%)	**13.08** (+23.9%)	**9.76** (+5.1%)	**17.48** (+31.7%)	**7.38** (+22.1%)	**7.36** (+37.0%)	**0.70** (+27.8%)	**1.01** (+17.9%)	**3.04** (+67.9%)	**7.63** (+37.5%)	**48.86** (+76.8%)
Qwen3-8B	One-Shot	7.76	21.78	14.10	41.44	11.65	14.02	12.12	12.30	9.83	17.16	-
	B-Alchemy (bo1)	**12.11**	31.50	13.86	39.47	9.62	14.47	5.22	6.44	6.62	16.98	5.81
	VIGA (bo1)	13.51 (-11.6%)	**27.07** (+14.1%)	**10.07** (+27.3%)	**28.12** (+28.8%)	**9.15** (+4.9%)	**12.24** (+15.4%)	**2.33** (+55.4%)	**2.58** (+59.9%)	**6.43** (+2.9%)	**11.53** (+32.1%)	**22.64** (+289.8%)
	B-Alchemy (bo4)	9.20	19.60	13.07	36.34	9.24	12.62	3.68	4.23	4.42	17.12	23.23
	VIGA (bo4)	**7.20** (+21.7%)	**17.97** (+8.3%)	**9.92** (+24.1%)	**24.28** (+33.2%)	**8.69** (+6.0%)	**10.08** (+20.1%)	**1.89** (+48.6%)	**2.53** (+40.2%)	**3.06** (+30.8%)	**9.30** (+45.7%)	**42.88** (+84.6%)

Table 3: Evaluation on BlenderGym. We report Photometric Loss (PL) and Negative-CLIP Score (NC). Here bo$N$ denotes the best-of-N setting.

Model	Setting	Exec. ↑	Qual. ↑	Over. ↑
GPT-4o	One-Shot	0.92	60.7	55.8
	VIGA	**0.95** (+3.3%)	**61.4** (+1.2%)	**58.3** (+4.5%)
Claude-3.5	One-Shot	**0.93**	65.3	60.7
	VIGA	0.90 (-3.2%)	**69.2** (+6.0%)	**62.2** (+2.5%)
Gemini-2.5	One-Shot	0.61	67.9	41.4
	VIGA	**0.69** (+13.1%)	**70.9** (+4.4%)	**48.9** (+18.1%)
Qwen3-8B	One-Shot	0.10	60.5	6.0
	VIGA	**0.54** (+440%)	**60.8** (+0.5%)	**32.8** (+446.7%)

Table 4: Evaluation on SlideBench. We report Execution, Quality, and Overall scores.

PL: photometric loss (RGB diff.)

N-CLIP: negative CLIP (CLIP-space similarity -ve)

VLM: VLM Score (use another VLM to assess)