

MethodAndAnalyze (System2-Focused, Challenge-Driven)

> InternVLAN System2
InternVLAN

- 3 -

0. Intro

Challenge A: Viewpoint-Induced Reuse Failure

- Intro 问题解决方法 token
 - 问题解决 position-wise matching 视觉-aware 问题解决

Challenge B: Instruction-Conditioned Semantic Drift

- Intro 介绍到“刷新”
 - instruction-guided refresh 指令引导刷新

- - -

1. **Analyze**

→ □□□□□□□□□□“□□□□□”□□□“□□□□”□

1.1 Challenge A

A-Analysis-1: Token ████-███████████

- `[[token]]` token[]
 - `[[token]]/[[token]]`
 - `[[token]]`
 - `[[token]]`

A-Analysis-2: “□□□□ vs □□□□□□” □□□□

- $\text{align}_{\text{baseline}}$
 - Baseline \square position-wise matching
 - Ours-Align \square view-aware alignment + matching
 - $\text{align}_{\text{baseline}}$ token $\text{align}_{\text{baseline}}$ Recall@Reusable \square

- Ours-Align ဗိုယ်အတွက်

A-Analysis-3: ဗိုယ်အတွက်

- ဗိုယ်အတွက် ပေါ်လောက်ခဲ့သူများ
- ၂၁ “ဗိုယ်အတွက် ပေါ်လောက်ခဲ့သူ” ရှင်း

1.2 Challenge B ဗိုယ်အတွက်

B-Analysis-1: ဗိုယ်အတွက်

- ဗိုယ် “ဦး-ဗိုယ်” ရှင်းဆို cross-attention/grounding score ဗိုယ်
- ဗိုယ်အတွက် landmark ရှင်းဆို
- ဗိုယ်အတွက်/ဗိုယ်အတွက်

B-Analysis-2: ဗိုယ် vs ဗိုယ်အတွက်

- ဗိုယ်
- Fixed refresh ၏ K ဗိုယ်
- Semantic refresh ဗိုယ်အတွက်
- ဗိုယ်အတွက် + ဗိုယ်
- ဗိုယ် Semantic refresh ဗိုယ်အတွက်

B-Analysis-3: ဗိုယ်အတွက်

- ဗိုယ်အတွက်/ဗိုယ်အတွက် ပေါ်လောက်ခဲ့သူ vs ဗိုယ်အတွက်
- ဗိုယ်အတွက်

2. ဗိုယ်Method ဗိုယ်အတွက်

> ဗိုယ်အတွက် **VLN-Cache-S2** System2-only enhancement

2.1 Method-A: View-Aware Reuse Module ဗိုယ် Challenge A

ဗိုယ်

- System2 ၏ token ဗိုယ်အတွက် view-aware ဗိုယ်

- token reuse

内部のルートを再利用

- A1: token reuse
- A2: depth/ルートの深さ depth + ルート
- A3: token reuse

System2 内部のInternVLA

- `internnav/model/basemodel/internvla_n1/internvla_n1_policy.py`
- `s2_step(...)` の “” → ルート -> ルート “”
- `internnav/agent/internvla_n1_agent.py`
- `step(...)` の ルート

2.2 Method-B: Instruction-Guided Semantic Refresh チャレンジ B

刷新

- refresh “” “”
- instruction-grounded saliency ルート

内部のルートを再利用

- B1: token reuse/ルート + ルート
- B2: cross-attention/ルート
- B3: token reuse “”

System2 内部のInternVLA

- `internnav/agent/internvla_n1_agent.py`
- `should_infer_s2(...)` の “” ルート “” + ルート “”
- `internnav/model/basemodel/internvla_n1/internvla_n1_policy.py`
- `s2_step(...)` の saliency ルート drift score

2.3 Method-C: Decode Budget Controller System2 内部のルート

□□□□

- □□□□/□□□□□□ `max_new_tokens` □□□□□□□□□
- □□□□□□□□□□□□□□□□□□□□

□□□□□□

- C1□□□□□□□□□□□□□□□□□□□□
- C2□□□□□□□□ draft □□□□speculative-lite□□
- C3□□□□draft+verify □□□ speculative decoding□

System2 □□□□InternVLA□

- `internnav/model/basemodel/internvla_n1/internvla_n1_policy.py`
 - `generate(...)` □□□□□□□□
- `scripts/eval/configs/habitat_dual_system_cfg.py`
 - □□□□□□□□□□□□□□□□□□□□

3. □□□□□□□□“□□□□”□□□□□□□□

3.1 □□ S□□□□

- □□□A1 + B1 + C1
- □□□□□□□□□□□□□□□□□□
- □□□□□□□□
- □□□□□□□□□□□□□□□□□□

3.2 □□ M□□□□□□□□

- □□□A2 + B2 + C1/C2
- □□□□-□□□□□□□□□□□□□□
- □□□□□□□□□□□□□□
- □□□IROS □□□□

3.3 □□ X□□□□

- □□□A3 + B3 + C3
- □□□□□□□□□□□□□□

- 𠂔/𠂔/𠂔/𠂔
- 𠂔/𠂔/𠂔/𠂔

4. 𠂔/𠂔/𠂔/𠂔

4.1 𠂔/𠂔/𠂔/𠂔

1. **Fig.1 Challenge Overview** A/B 𠂔/𠂔/𠂔/𠂔
2. **Fig.2 Viewpoint Mismatch Analysis** 𠂔/𠂔 vs 𠂔/𠂔/𠂔/𠂔
3. **Fig.3 Semantic Drift Analysis** 𠂔/𠂔/𠂔/𠂔 + refresh 𠂔/𠂔
4. **Fig.4 Method Pipeline** A/B/C 𠂔/𠂔 System2 𠂔/𠂔
5. **Fig.5 Pareto Curve** 𠂔/𠂔-Latency vs SR/SPL 𠂔/𠂔

4.2 𠂔/𠂔/𠂔/𠂔

1. **Tab.1 Main Results** NE/OS/SR/SPL + 𠂔/𠂔 System2 𠂔/𠂔
2. **Tab.2 Ablation by Challenge** Base+A+B+A+B+A+B+C
3. **Tab.3 Overhead Breakdown** 𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔

5. 𠂔/𠂔/𠂔/“𠂔/𠂔”

5.1 𠂔/𠂔/𠂔/𠂔

- E0: Baseline 𠂔/𠂔 System2 𠂔/𠂔
- E1: Baseline + Method-A 𠂔/𠂔 Challenge A 𠂔/𠂔
- E2: Baseline + Method-B 𠂔/𠂔 Challenge B 𠂔/𠂔
- E3: Baseline + A + B 𠂔/𠂔/𠂔/𠂔/𠂔/𠂔
- E4: Baseline + A + B + C 𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔

5.2 𠂔/𠂔/𠂔/𠂔

- 𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔/𠂔
- 𠂔/𠂔/𠂔/𠂔/“/token”
- 𠂔/𠂔/𠂔/𠂔/“SR/SPL/NE”

5.3 问题

- SR, SPL, NE, OS
- System2 问题 tokenS2 问题
- A 问题B 问题/问题

6. 问题

1-2 问题

- C1+ B1
- 问题

2-3 问题

- A2+ B2
- Fig.2 / Fig.3 / Tab.2

3 问题

- C2 C3 speculative 问题

7. 问题 claim 问题

- “System2 问题”“问题”“问题”
- “layer-adaptive reuse” 问题/问题
- claim 问题

8. 问题

1 问题

“For Challenge A, we introduce a view-aware reuse module that aligns cross-frame representations before token reuse, mitigating viewpoint-induced mismatch.”

□□□ **2**□□□□□□□

“For Challenge B, we design an instruction-guided semantic refresh mechanism that updates cached regions only when task relevance shifts, reducing stale-cache errors.”

□□□ **3** □□□□□

"All modules are integrated into the System2 branch of InternVNL, leaving training setup unchanged and focusing on inference-time acceleration."

1

- **“ $A_2 + B_2$ ”** $\leftrightarrow M: A_2 + B_2 + C_1/C_2$
 - **System 2**
 - 1) **intro** **System 2**
 - 2) **System 2** **System 2**
 - 3) **System 2** **System 2**