Vision-Language-Based Social Navigation Decision on Unitree Go1

Yuni Wu

CSCI 7000 Vision-Language Models in Robotics

Motivation

Robots operating in human environments need **more than obstacle avoidance.**

People move with intent.
Understanding motion and direction is essential for safe and natural navigation.

Context matters.
The same scene can require different actions based on social context.

Safety and trust.
Socially-aware decisions lead to safer and more acceptable robot behavior.

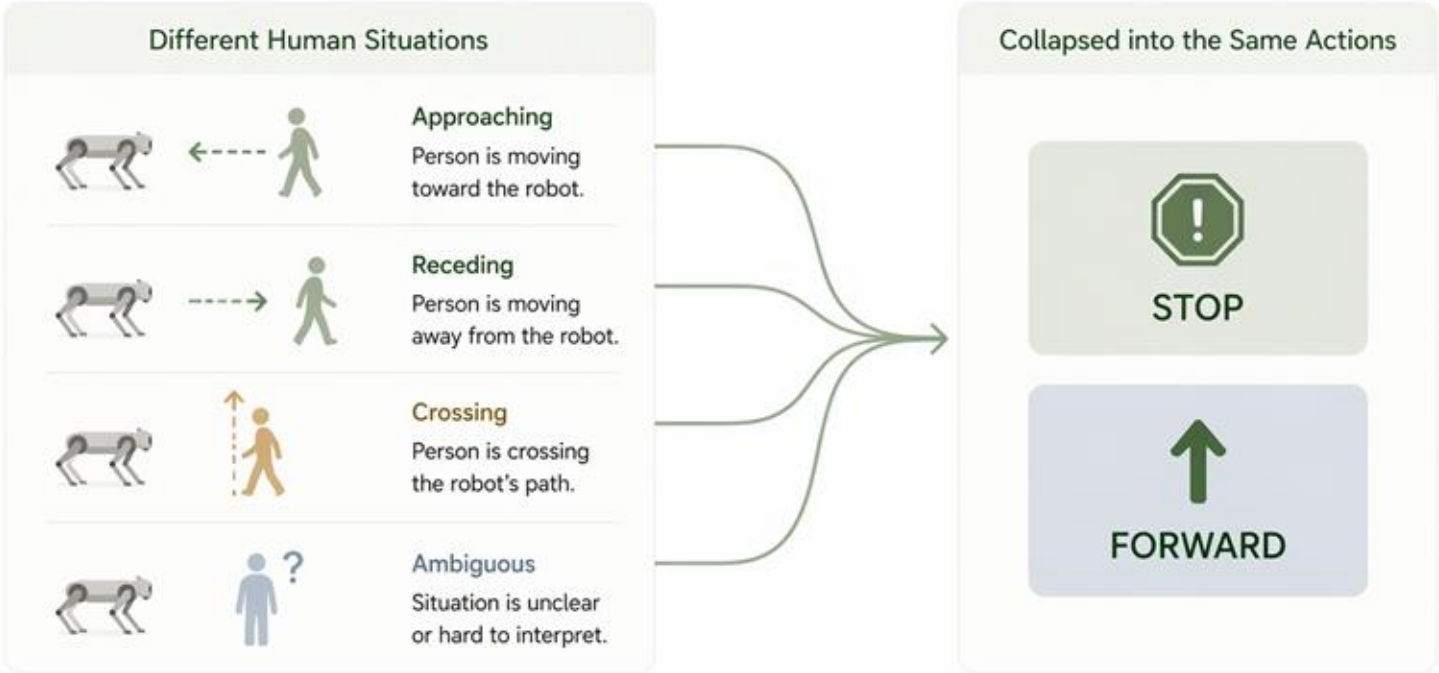
Can a pretrained VLM,
with only prompt-level intervention,
serve as a social navigation decision maker?

Example Corridor Scenarios

Human-aware navigation requires understanding people, not just avoiding obstacles.

The Problem

Different human situations, but the **same robot action.**



Socially different situations → identical decisions

The robot cannot express meaningful social behavior.

Idea

I use **pretrained VLMs** to produce **structured social navigation** decisions.

1 Perception

Image / Sequence

Recent frames from robot camera

2 VLM Reasoning

Pretrained VLMs

Qwen3-VL-30B

InternVL-3.5-14B

I leverage powerful VLMs to understand motion, intent, and context using world knowledge.

3 Structured Output

Structured Decision

```
{
  "motion": "crossing_rightward",
  "crossing_direction": "right",
  "recommended_avoidance_side": "left",
  "recommended_action": "LEFT",
  "reasoning": "Person is crossing
  from right to left in front of
  the robot."
}
```

I produce consistent JSON output with reasoning.

4 Action Space

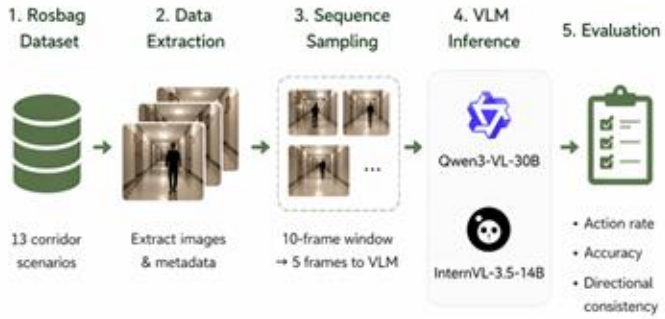
- STOP
- FORWARD
- LEFT
- RIGHT
- REVIEW (Ambiguous)

My goal is to build a **human-aware navigation** system that understands people and acts appropriately.

System Overview

Two pipelines share the same decision logic:
offline benchmark for evaluation, real-time controller for deployment.

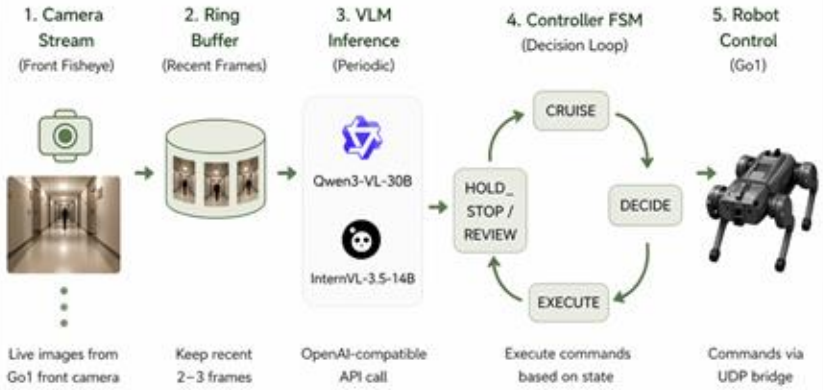
1 Offline Benchmark (Evaluation)



Output

★ Total: 1,173 sequence probes across 13 bags
5 images per probe (sampled from a 10-frame window)

2 Real-Time Controller (Deployment)



Controller Executable Actions (Safety-Preserving Projection)

Advisory (Not Executed)

Saved as avoidance hints for future geometric checks

Key Point: Both pipelines use the same decision logic and VLMs.
The difference lies in data source, timing, and how outputs are used.

- **Offline:** for evaluation and analysis
- **Real-time:** for safe deployment on Go1

Structured Rules

Simple and interpretable policy for robot navigation

Forward when safe

Path is clear or person is moving away.

→ FORWARD

Avoid when crossing

Person is crossing ⟶ move to the opposite side.

→ LEFT / RIGHT

Review when unsure

Situation is ambiguous or unclear.

→ REVIEW

Stop when blocked

Path is blocked or no safe option.

→ STOP

These rules help the robot make safe and interpretable decisions.

Benchmark & Results

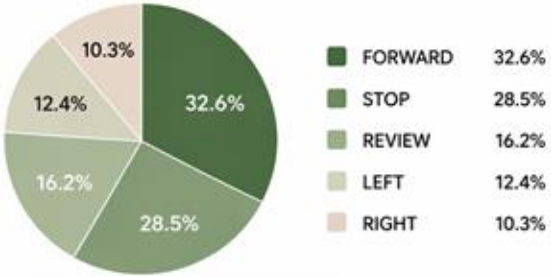
Benchmark Setup

13
rosbags
(corridor scenarios)

1,173
sequence probes
(5-frame input)

5 actions
STOP / FORWARD /
LEFT / RIGHT / REVIEW

Action Distribution (All Probes)



The model actively uses all actions,
not limited to STOP and FORWARD.

Key Findings

Non-collapsed actions
The model uses all 5 actions across different social situations.

Directional consistency = 1.000
When the model decides LEFT/RIGHT in crossing scenarios,
the direction is always correct.

REVIEW behavior is observed
The model requests more information in ambiguous situations.

VLMs can form structured, socially-aware navigation policies.

Key Takeaway

The policy capability exists.
The model understands people's motion and intent,
and can make appropriate social navigation decisions.

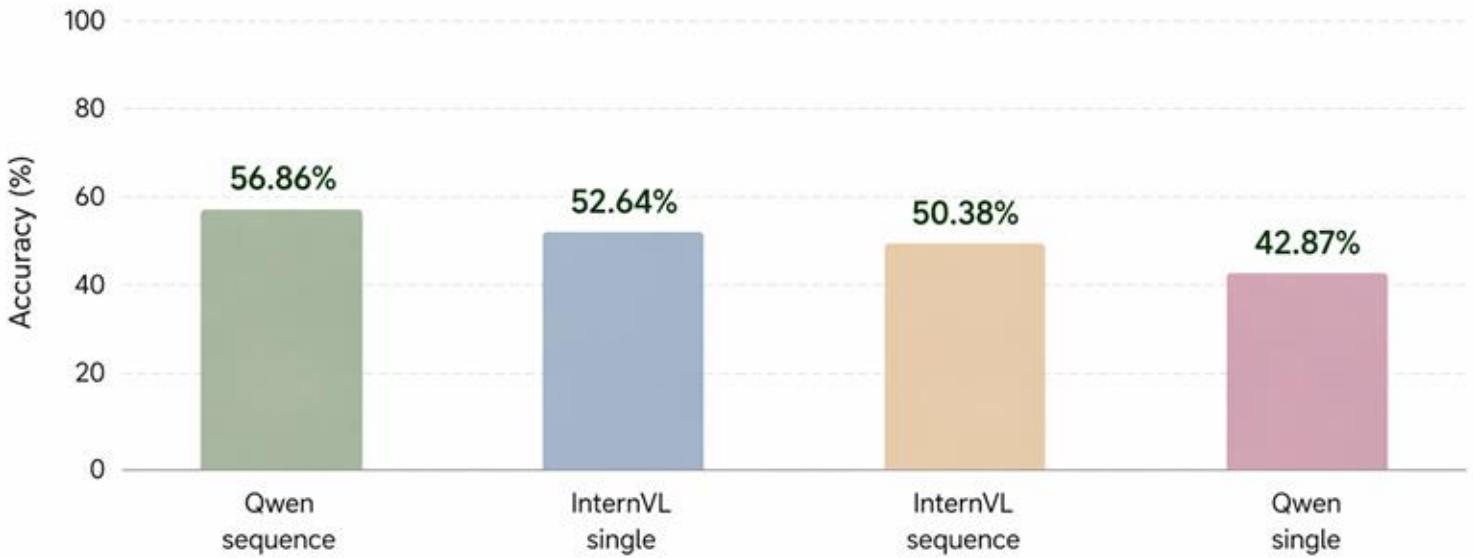
But failures still occur.
Some scenarios are still misclassified or ignored.
Rule correctness ≠ Rule activation.

Method Comparison (Sample-level Accuracy)

Per-frame accuracy across the 13 bags (1173–1290 frames)

Overall Sample-level Accuracy

(% of all frames)



Key Takeaway

Sequence methods achieve higher overall sample-level accuracy than single-image methods.

Takeaway & Future Work

Takeaway

Structured rules with VLMs work.
The system can make directional decisions and handle uncertainty.

Main limitation is perception.
Crossing events are often not detected, so rules are not triggered frequently enough.

Policy is correct,
but the model does not see the right situations often enough.

Future Work

- Collect more real-world data.
- Fine-tune the model to improve perception, especially motion understanding.
- Improve real-time perception quality and reduce latency.
- Explore better temporal modeling.

The policy is correct, but the model does not see the right situations often enough.