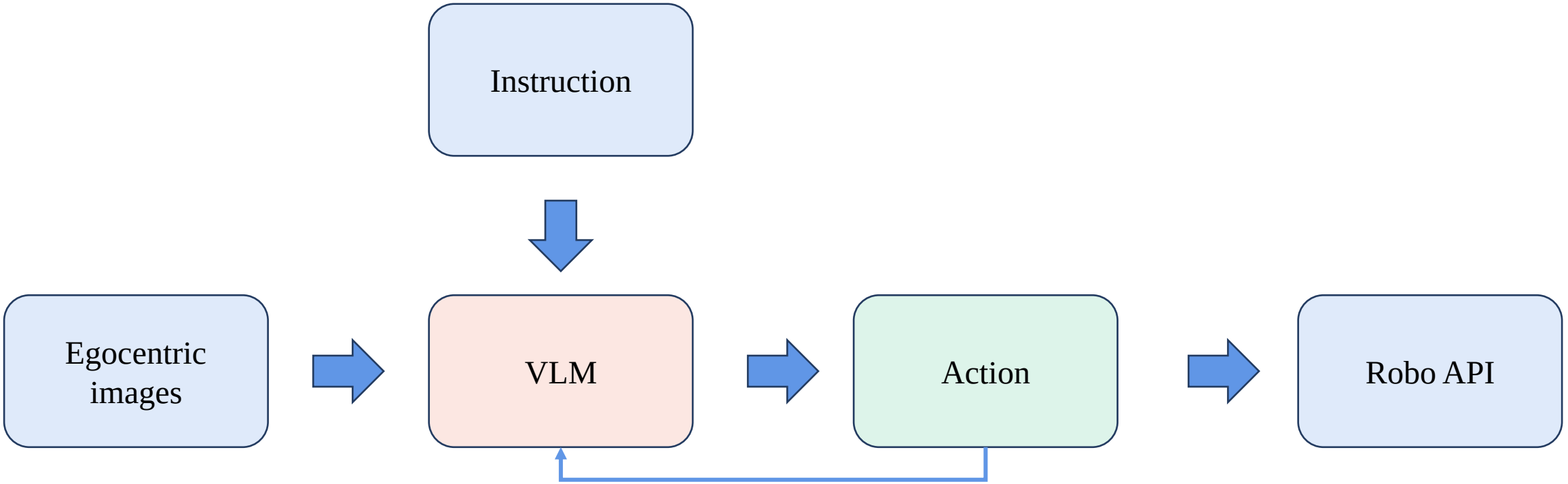
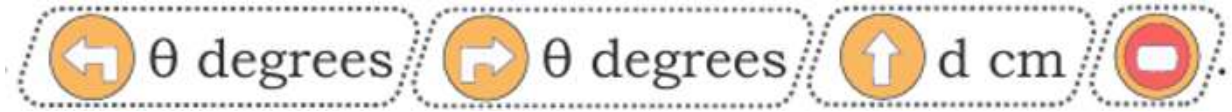
VLN



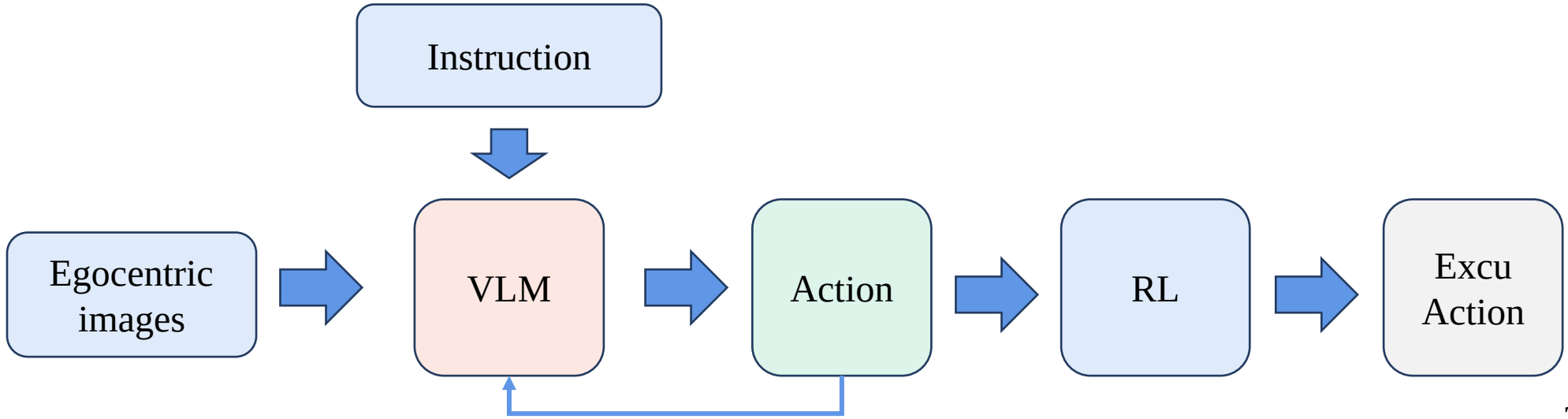
Action

Loose movement space

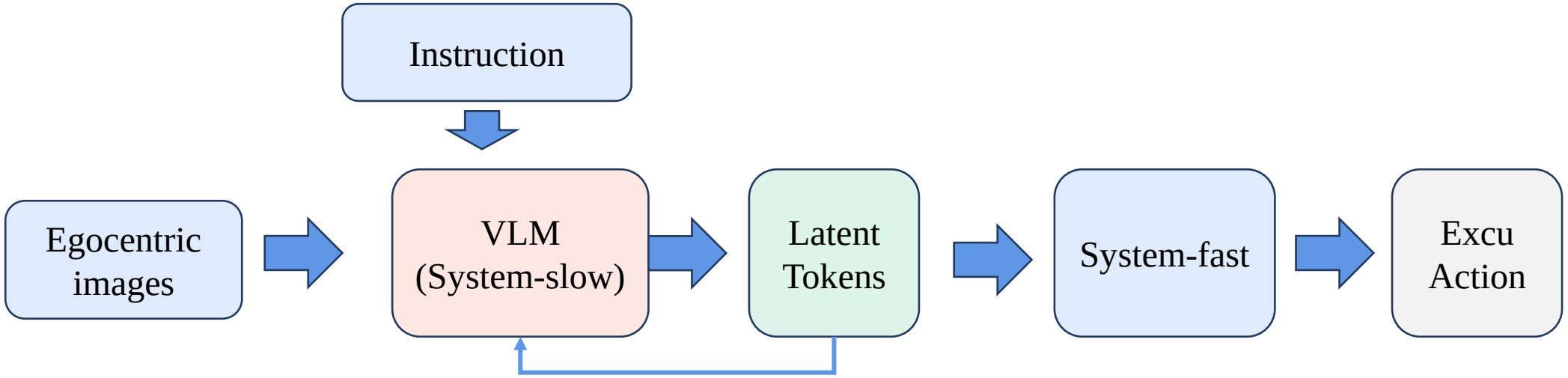
fixed movement space



VLN



2025.5



2025.9

NaVILA (RSS 2025)

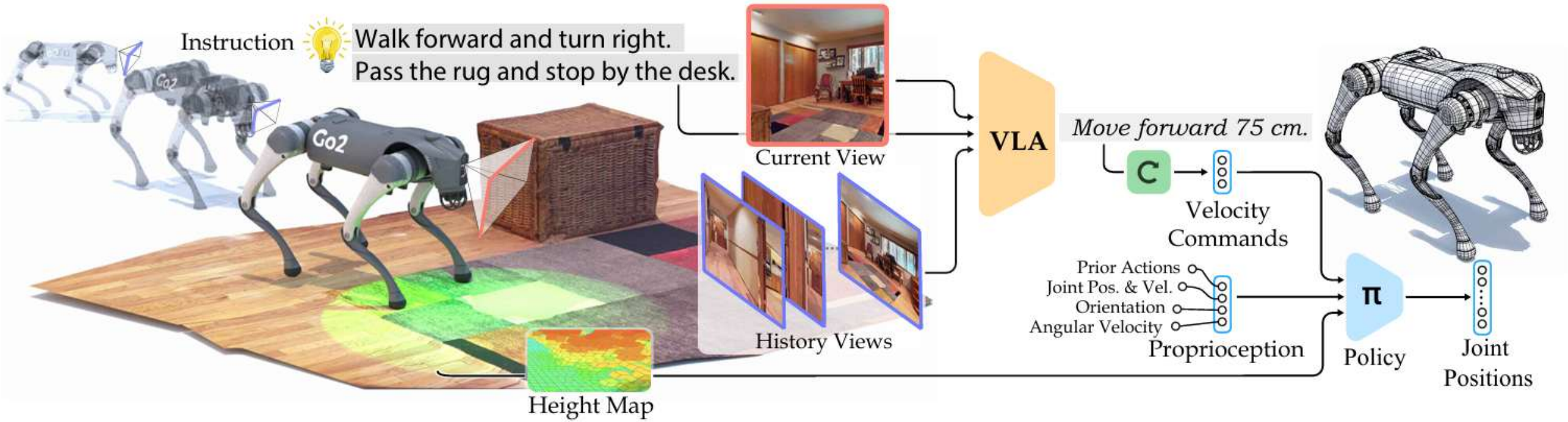


Fig. 2: NaVILA is a two-level framework combining high-level visual language understanding with low-level locomotion control. Our VLA model processes single-view images to produce mid-level actions in natural language, which are then converted into precise joint movements by an advanced low-level locomotion policy. This integration allows for strong generalization and adaptability across different real-world environments, and can operate the robot in real-time.

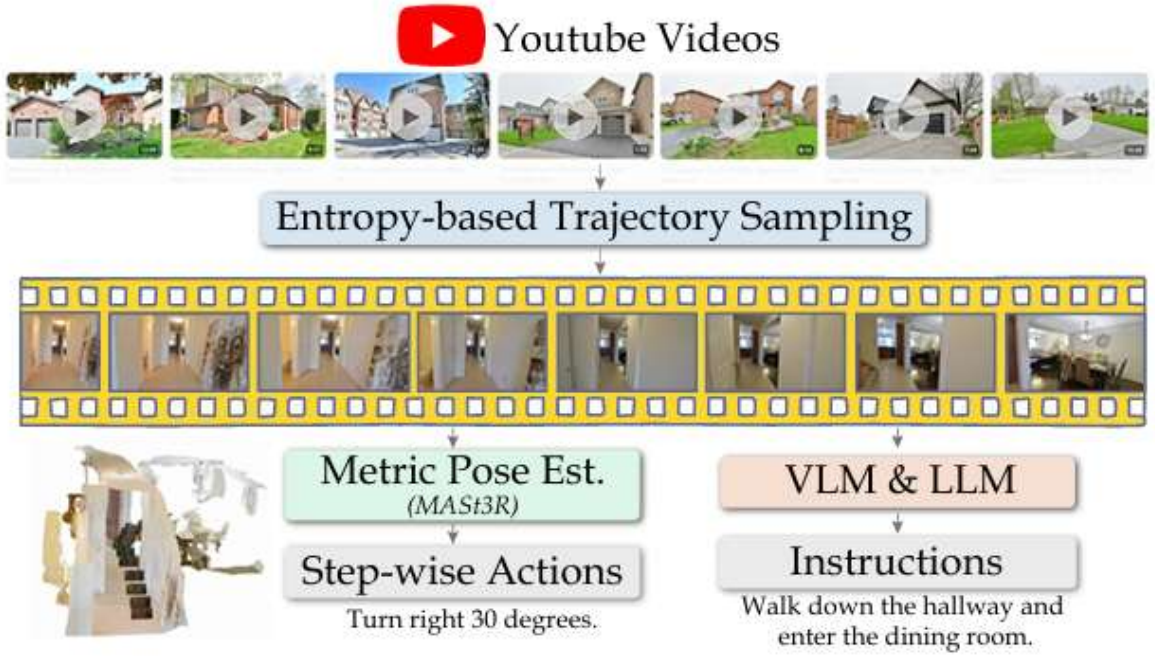


Fig. 4: Data pipeline for transforming human touring videos in the wild into pairwise navigation data within a continuous environment. We begin by processing the videos into meaningful trajectories through entropy-based sampling [26]. Then we extract step-wise actions through metric camera pose estimation [27], and utilize VLM [13] and LLM [28] to generate instructions.

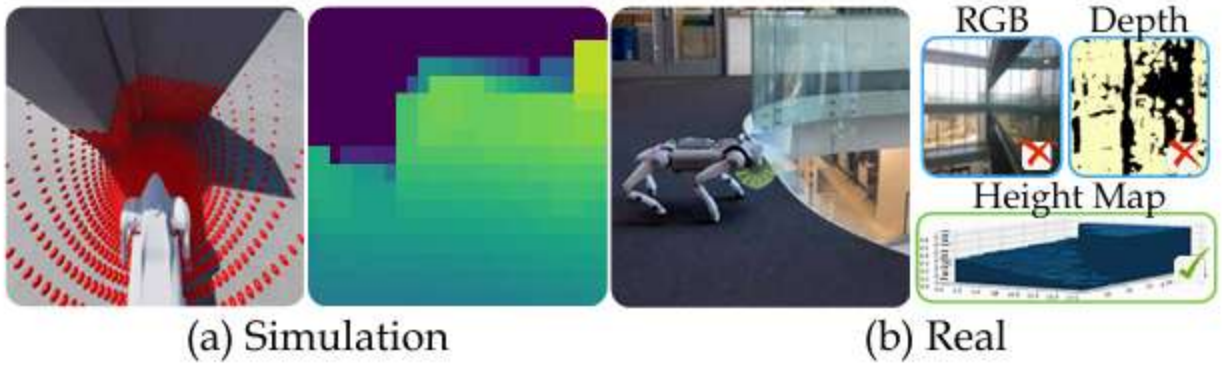


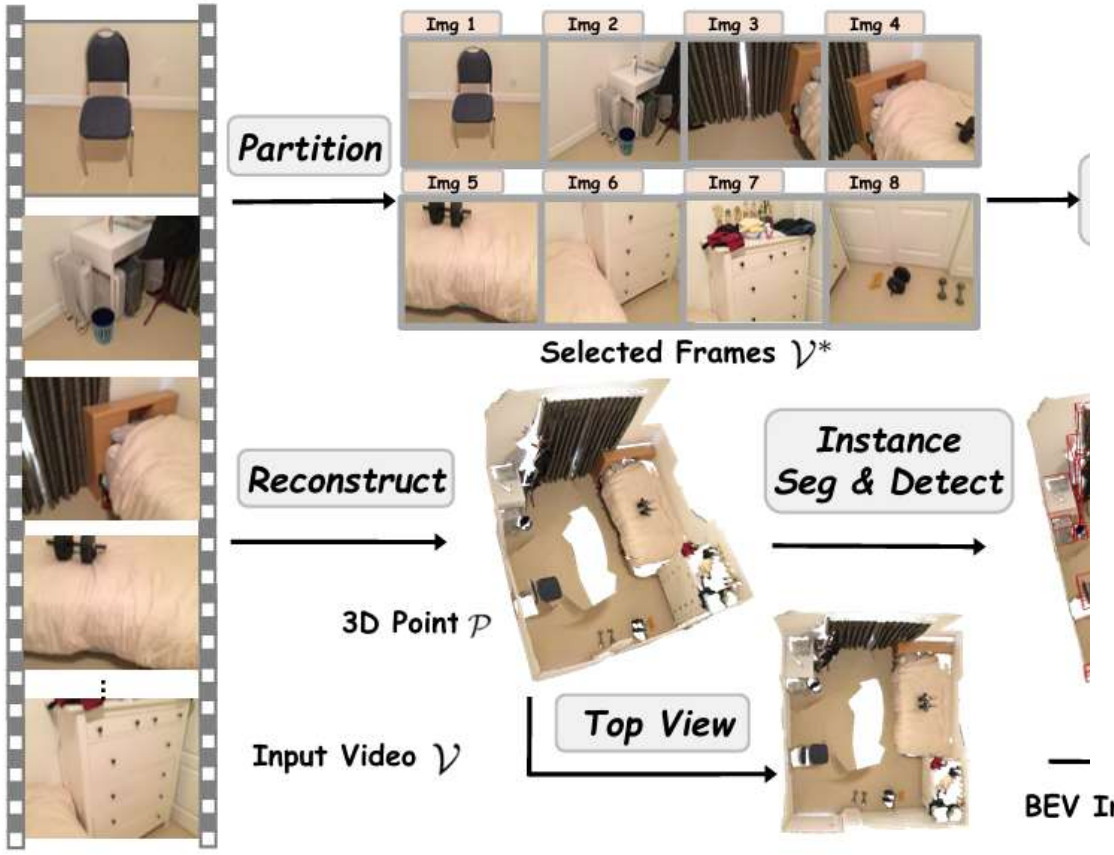
Fig. 5: Height map reconstruction from point cloud. (**a**) Go2 robot follows velocity commands while avoiding obstacles in simulation. Red dots show LiDAR points raycasting from the sensor center to the terrain mesh. The right image shows a preprocessed height map with values clipped to sensor constraints; darker colors indicate higher heights. (**b**) Safe locomotion near glass. The top-down height map detects glass surfaces where depth and RGB images fail.

What skills do VLM need to learn?

1. 3D scene awareness of 2D images
2. Long and short term memory
3. Positioning and planning navigation capability
4. Action space
5. Exploration and error correction

……

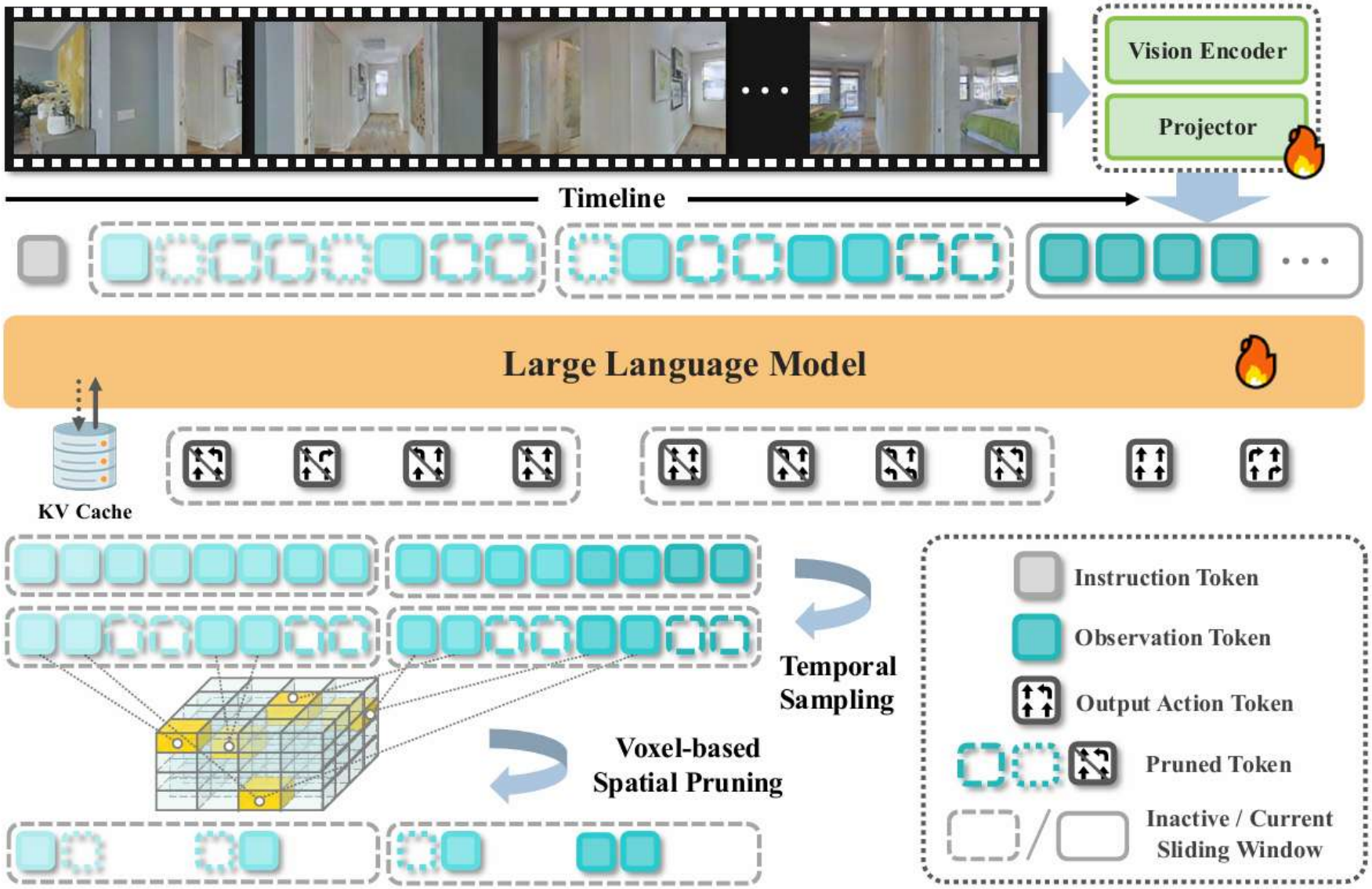
GPT4Scene: Understand 3D Scenes from Videos with Vision-Language Models (Arxiv 2501)



Zero-shot 3D QA	ROUGE@ScanQA		EM-1@SQA3D	
	VID	+GPT4Scene	VID	+GPT4Scene
Open-sourced VLM Based Model				
InternVL2-8B [22]	34.3	33.7 -0.6	33.0	31.4 -1.6
MiniCPM-V-2.6-8B [111]	31.5	32.1 +0.6	42.6	43.3 +0.7
Qwen2-VL-2B [96]	28.2	28.4 +0.2	35.7	34.8 -0.9
Qwen2-VL-7B [96]	29.3	31.7 +2.4	40.7	41.7 +1.0
Qwen2-VL-72B [96]	30.4	33.4 +3.0	39.8	42.3 +2.5
Closed-sourced VLM Based Model				
GPT-4o [72]	32.6	37.7 +5.1	40.3	42.8 +2.5
Gemini-1.5-Pro [88]	33.4	37.5 +4.1	41.7	44.2 +2.5
3D LLM Based Model	*Pre SOTA*		*Pre SOTA*	
Chat-Scene [39]	*41.6*		*54.6*	

- Most 7B models inherently lack 3D scene perception capabilities.
- Existing approaches enhance 7B scene perception through methods like fine-tuning with instructions and BEV.
- How to achieve 3D scene perception using only 2D images: Spatial Reasoning VLM.

StreamVLN（end-to-end）



1. Long and short term memory
2. Positioning and planning navigation capability
3. Exploration and error correction

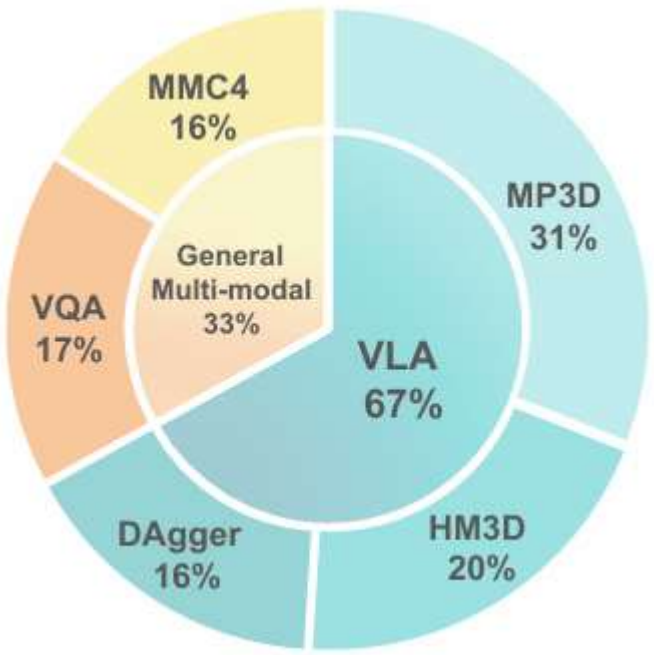
StreamVLN（end-to-end）

0~32

0~64

0~96

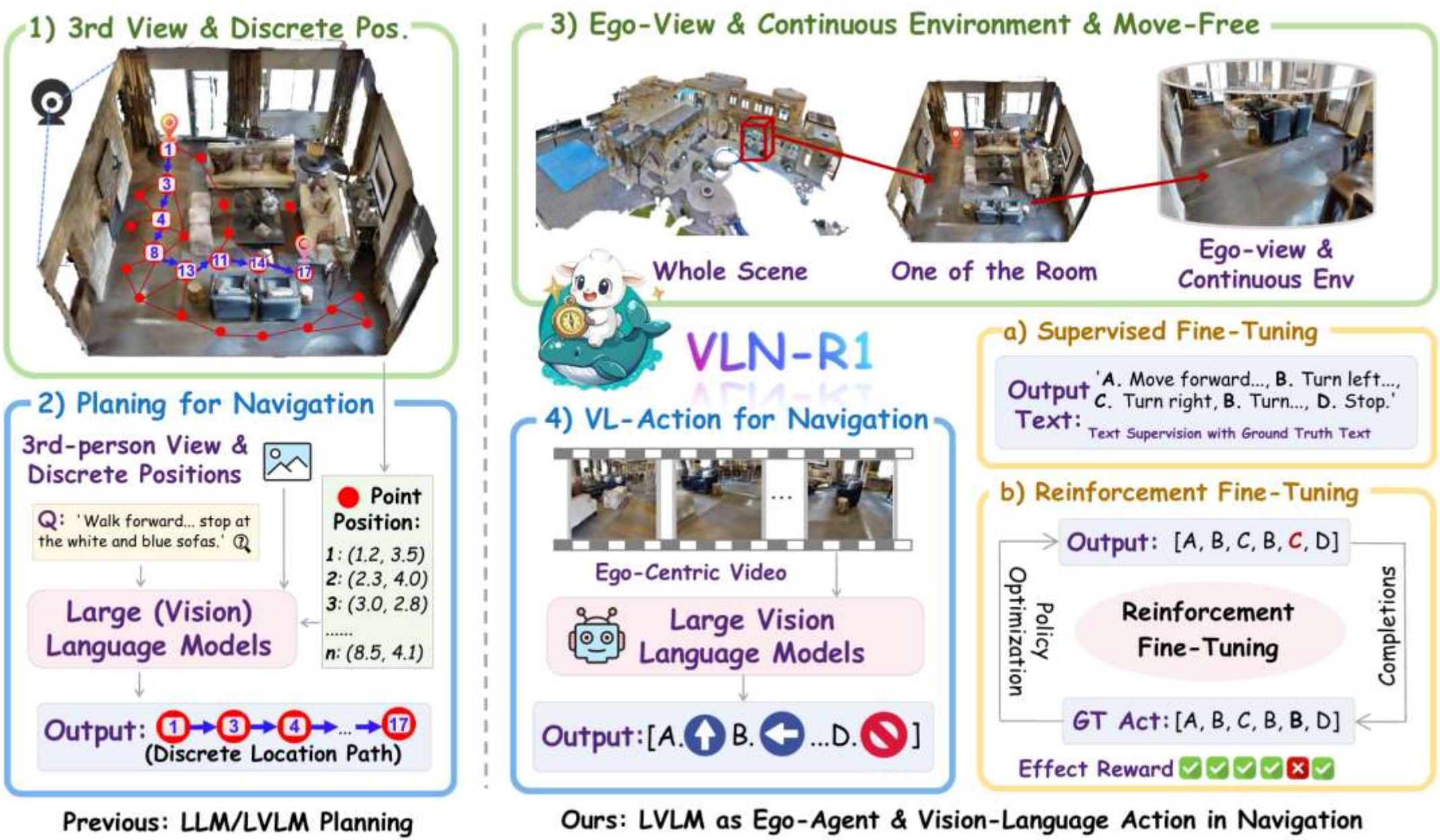
StreamVLN



Two-stage training:

1. Expert path SFT (input image sequence + instructions, output actions)
2. Collect the path of the first-stage model in the simulation environment and obtain the current expert action；
3. Dagger training + video question answering and graphic multi-round dialogue data set joint training

VLN-R1

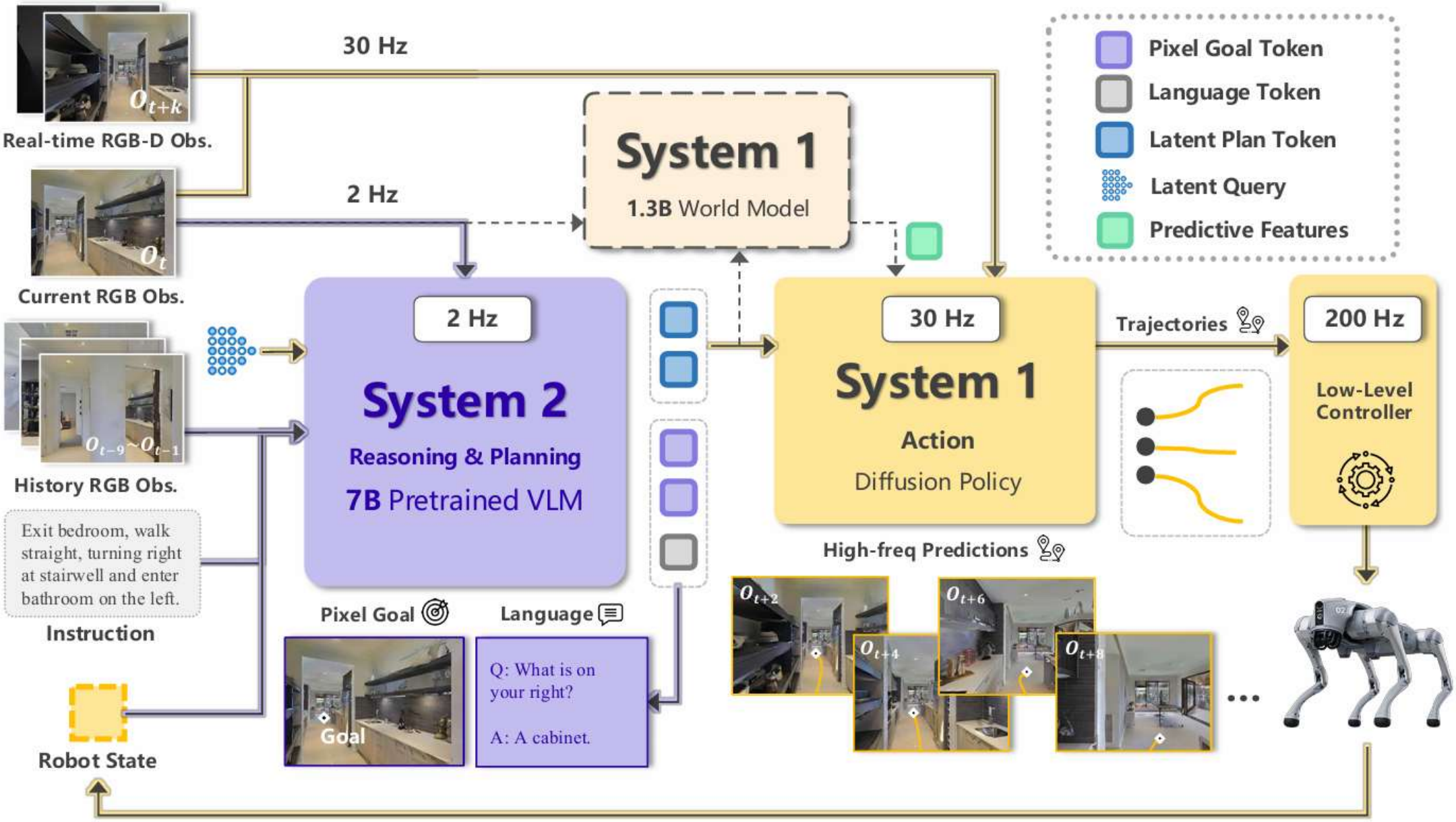

1) 3rd View & Discrete Pos.
2) Planing for Navigation
3rd-person View & Discrete Positions
Q: 'Walk forward... stop at the white and blue sofas.'
Large (Vision) Language Models
Point Position:
1: (1.2, 3.5)
2: (2.3, 4.0)
3: (3.0, 2.8)
.....
n: (8.5, 4.1)
Output: 1 → 3 → 4 → ... → 17
(Discrete Location Path)
Previous: LLM/LVLM Planning
3) Ego-View & Continuous Environment & Move-Free
Whole Scene
One of the Room
Ego-view & Continuous Env
VLN-R1
4) VL-Action for Navigation
Ego-Centric Video
Large Vision Language Models
Output: [A. B. ...D.]
a) Supervised Fine-Tuning
Output Text: 'A. Move forward..., B. Turn left..., C. Turn right, B. Turn..., D. Stop.'
Text Supervision with Ground Truth Text
b) Reinforcement Fine-Tuning
Output: [A, B, C, B, C, D]
Policy Optimization
Reinforcement Fine-Tuning
Completions
GT Act: [A, B, C, B, B, D]
Effect Reward
Ours: LVLM as Ego-Agent & Vision-Language Action in Navigation

Two-stage training for end-to-end VLN

Two-stage training：

1. Expert path SFT: generates actions based on frame sequences and instructions；
2. Phase 2: capability enhancement
    1. Multi-source joint training: cross-temporal and geometric reasoning (multimodal question answering), imitation learning, contextual understanding (multiple rounds of text-image dialogue)
    2. RL

Dual System – InternVLA-N1 (Arxiv 2509)



Dual System – InternVLA-N1

Task Decoupling：

System-1

A multi-objective executor based on **diffusion policies** generates high-frequency **short-term trajectories**. It takes a **latent plan** as input, generates a set of candidate short-term trajectories, scores them using a critic, and selects the optimal trajectory.；

- Positioning and planning navigation capability
- Action space

System-2(VLM)

Image-based **waypoint planning** using Qwen-VL-2.5-7B, taking video and instructions as input and outputting 2D pixel coordinates or latent plan tokens;；

- 3D scene awareness of 2D images
- Long and short term memory

Dual System – InternVLA-N1

Two-stage Training：

System-1

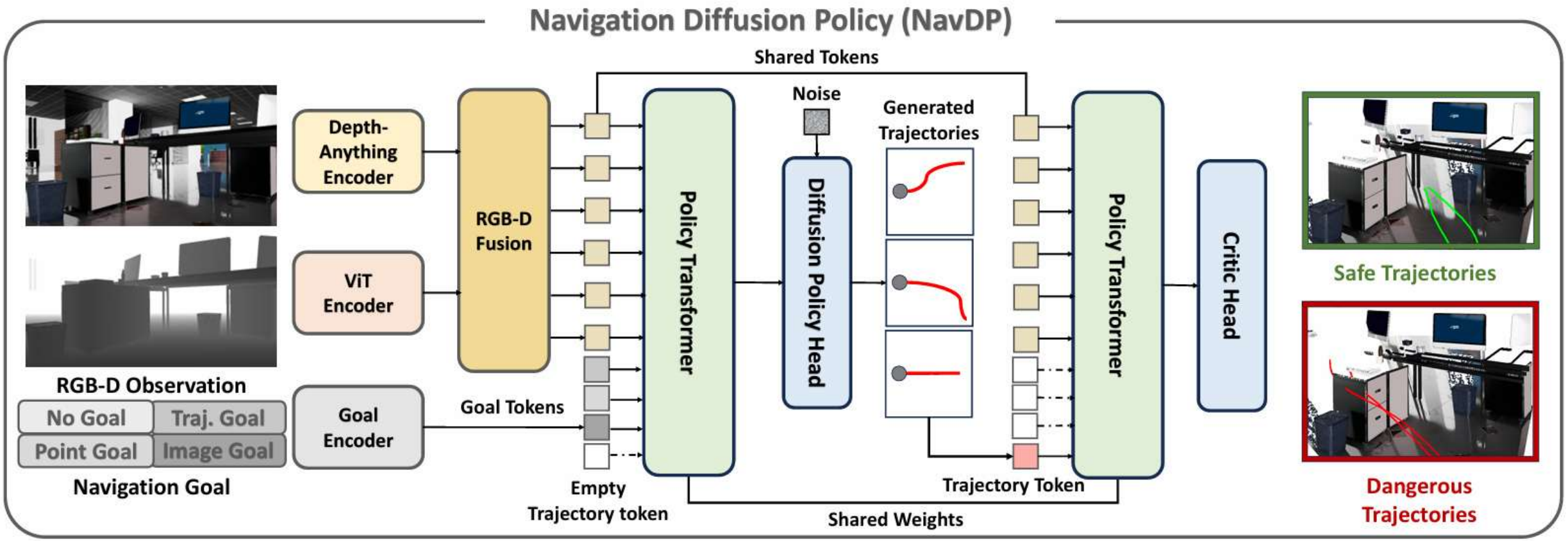
1. Embedding alignment among different goals
2. Diffusion Strategies with Noise Prediction
3. Critic prediction

System-2(VLM)

Training **pixel predictor**: The dataset includes commands, first-person observations, and labels for **mid-term navigation**; the point model learns command context understanding and predicts pixel-level goal locations within images；

Multi-system Joint Fine-tuning: Freeze the VLM, introduce **learnable latent queries** as both instruction fine-tuning and asynchronous inputs, and utilize **world-model** video prediction as an auxiliary task to enhance the spatial consistency of latent plans and reduce ambiguity in 2D pixels.

Diffusion Policy



Fuse the 256+256 tokens from RGB and Depth into 16 tokens, then construct the diffusion and critic models using the goal token and empty trajectory token.

Dual System – Diffusion Policy

Diffusion

生成 24-waypoints 的相对位姿序列（Δx, Δy, Δω）

输入：融合token，goal token，轨迹 token

DDPM scheduler逐步去噪生成轨迹

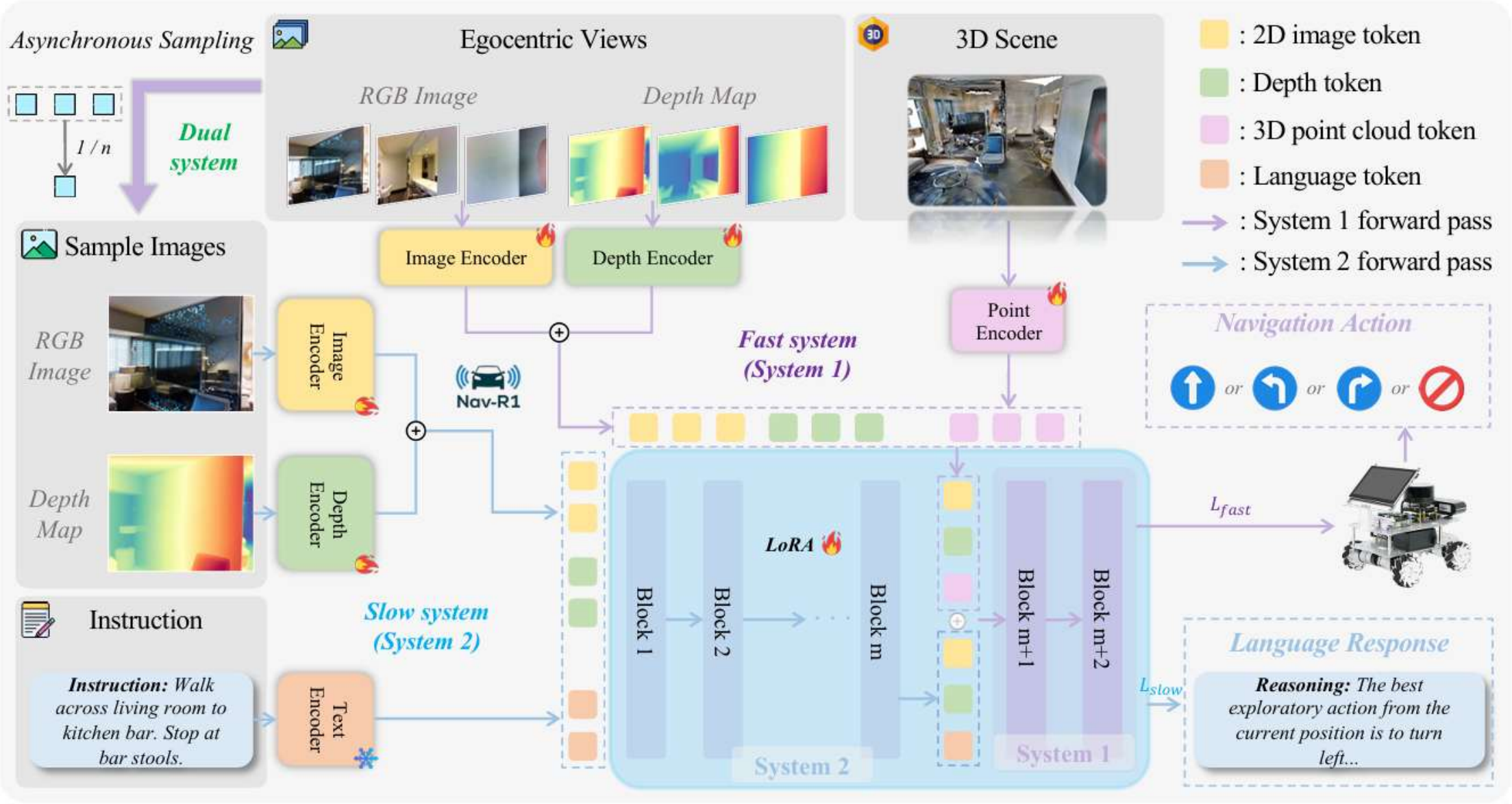
Critic

轨迹 token 经 1D conv 编码后与观测融合（不包含goal token），输出轨迹“安全价值”评分

训练：

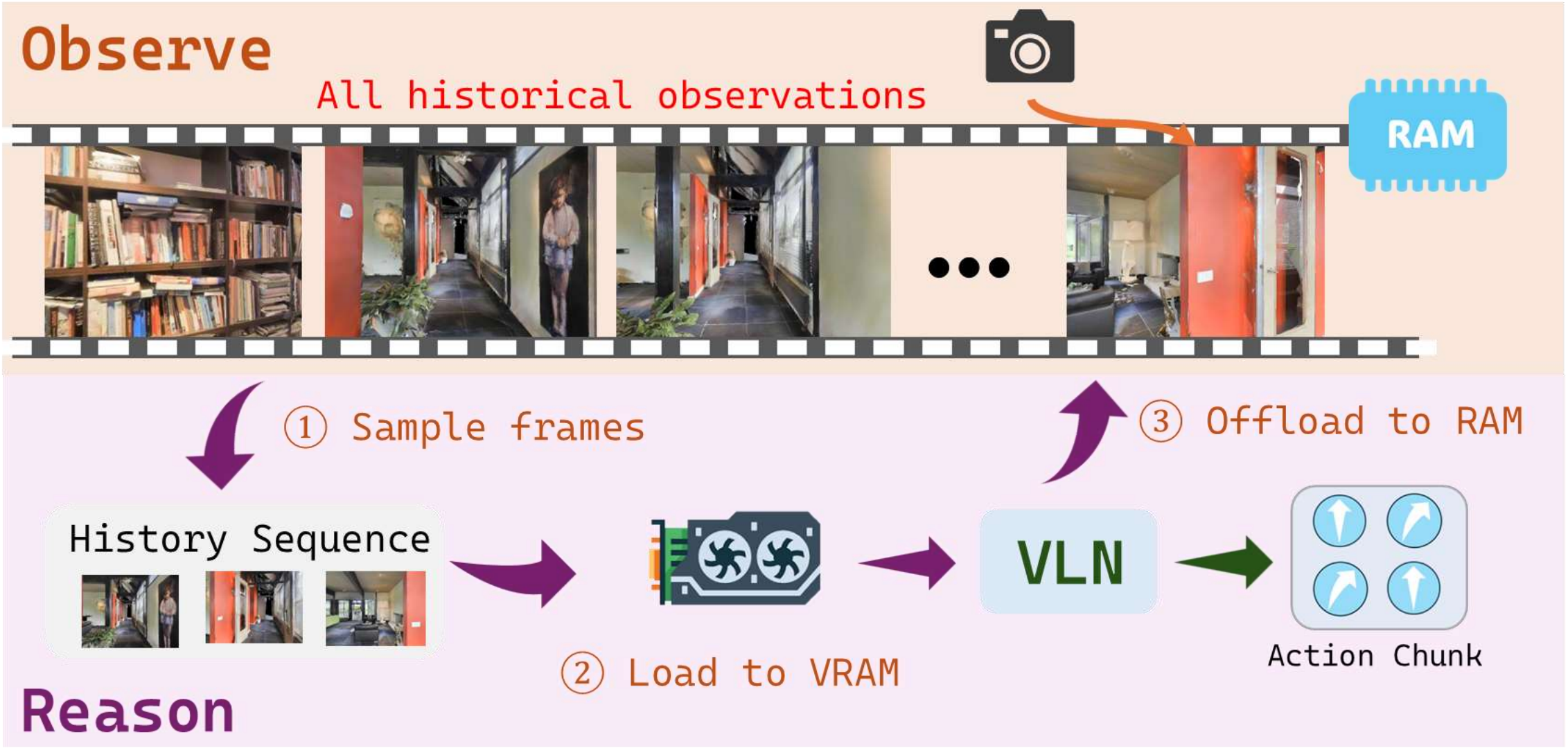
构造augmented / perturbed 轨迹，用模拟中的 ESDF（privileged global map）来计算每个 waypoint 到最近障碍物的距离，从而给每条轨迹构造一个标量价值标签

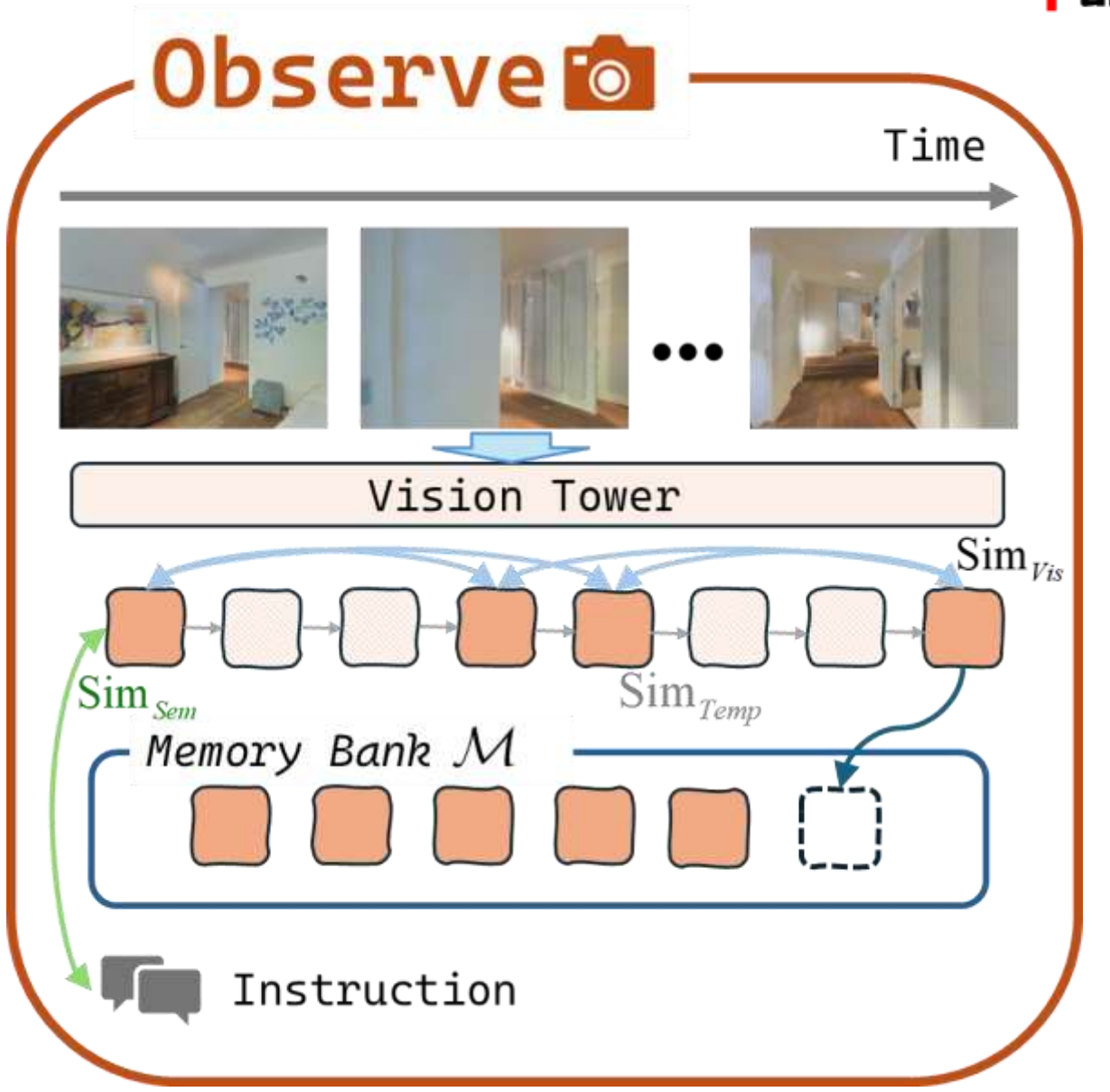
Dual System – Nav-R1 (Arxiv2509)

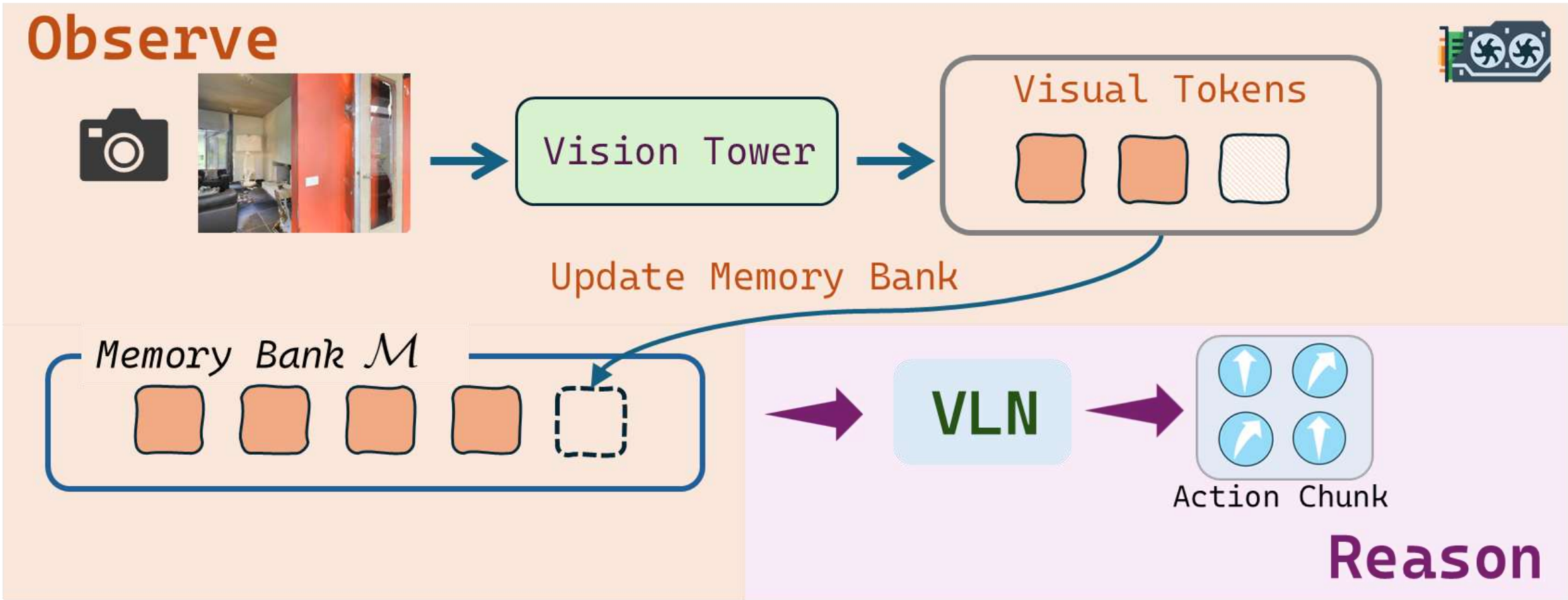


Why dual system?

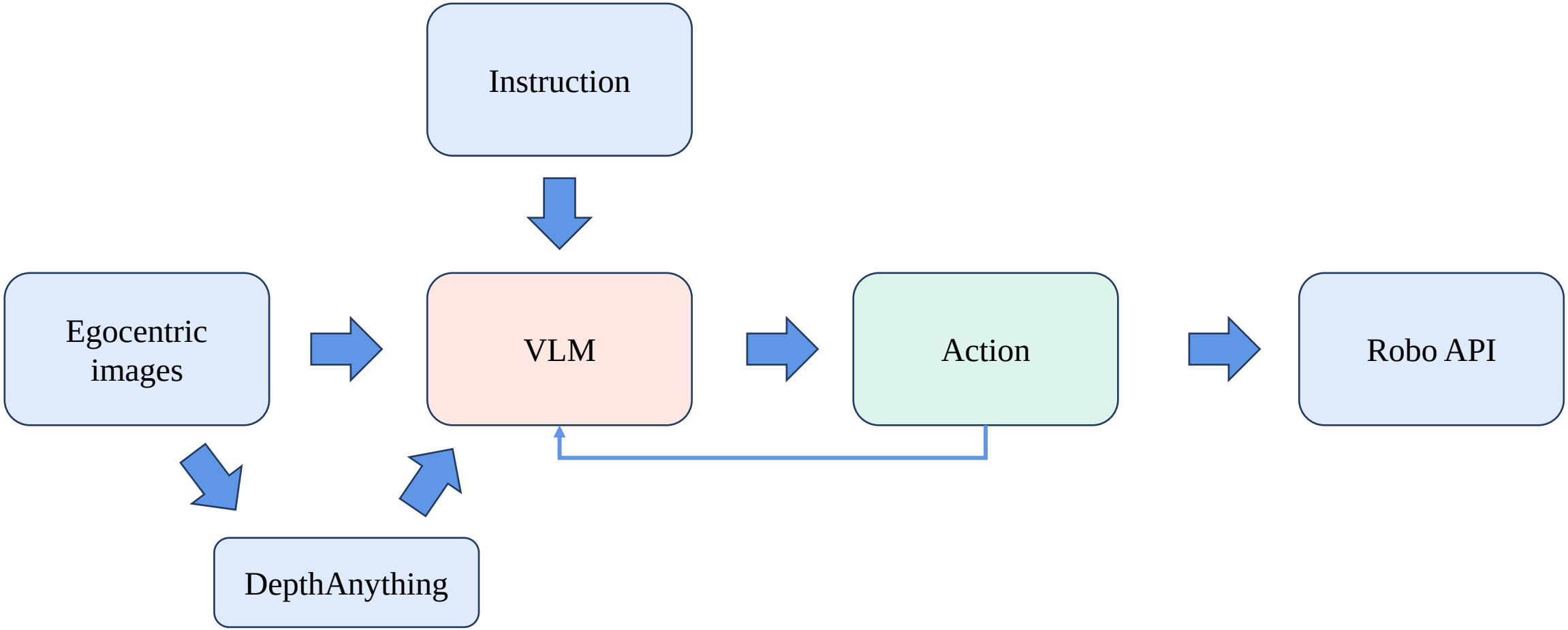
1. Task Decoupling: Focusing on spatial reasoning and instruction comprehension.
2. **Depth** information can be introduced.
3. Higher execution efficiency.


Observe
All historical observations
RAM
① Sample frames
③ Offload to RAM
History Sequence
② Load to VRAM
VLN
Action Chunk
Reason


Observe
Time
Vision Tower
Sim Vis
Sim Sem
Sim Temp
Memory Bank M
Instruction

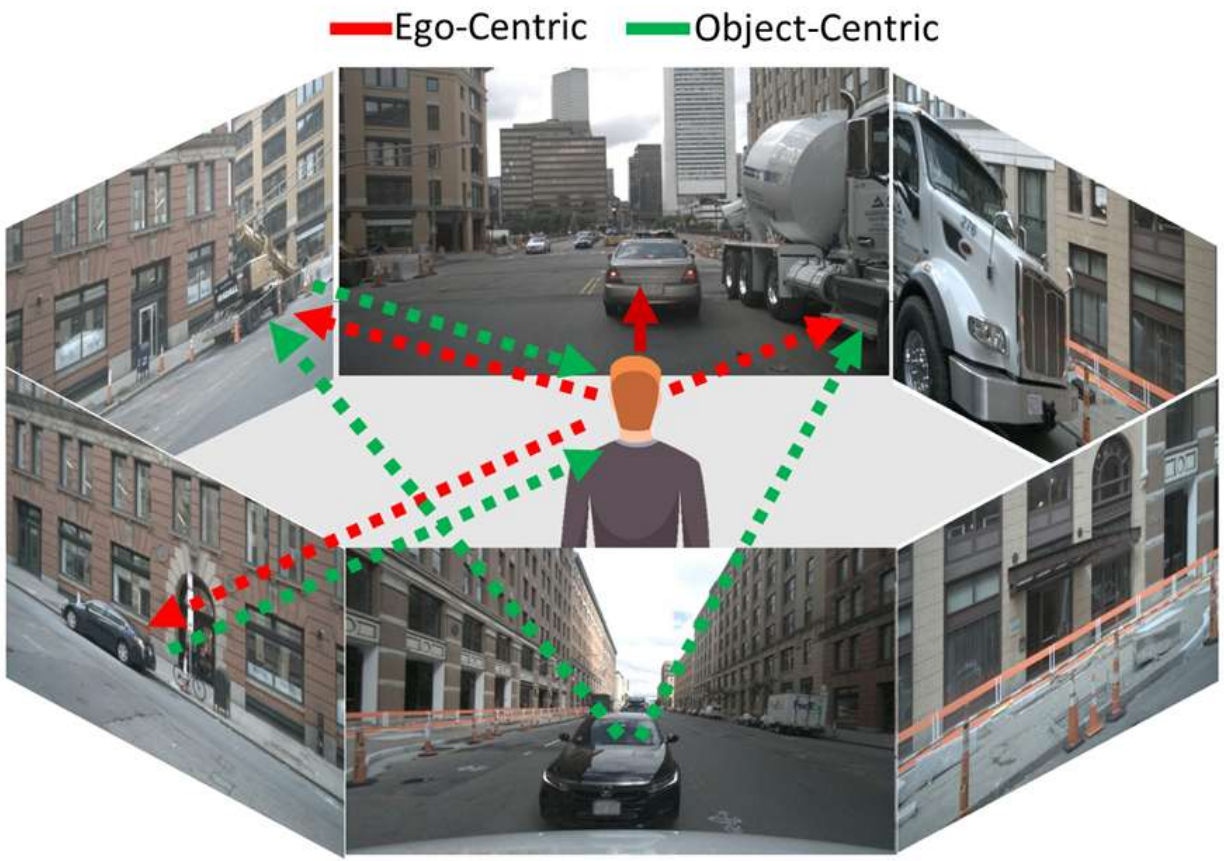

Observe
Vision Tower
Visual Tokens
Update Memory Bank
Memory Bank $\mathcal{M}$
VLN
Action Chunk
Reason

VLN - 3D Modeling



Spatial perception VLM training pipeline?

Spatial Reasoning with Vision-Language Models in Ego-Centric Multi-View Scenes (Arxiv 2511)



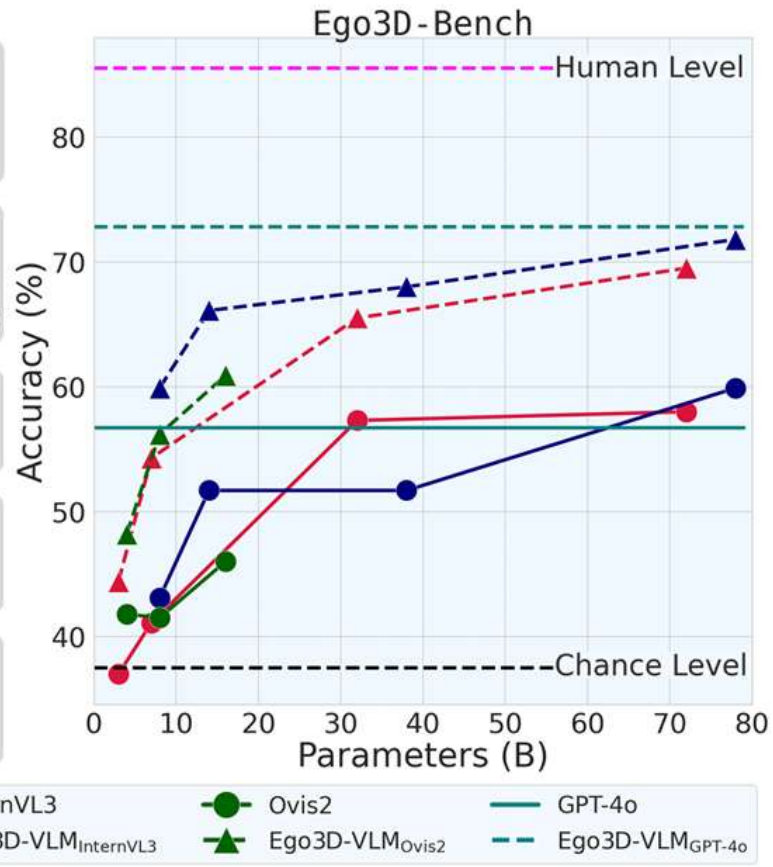
Absolute Distance: How far objects are cross-views?

Travel Time: What is the required time to reach a specific location?

Relative Distance: Which object is closer?

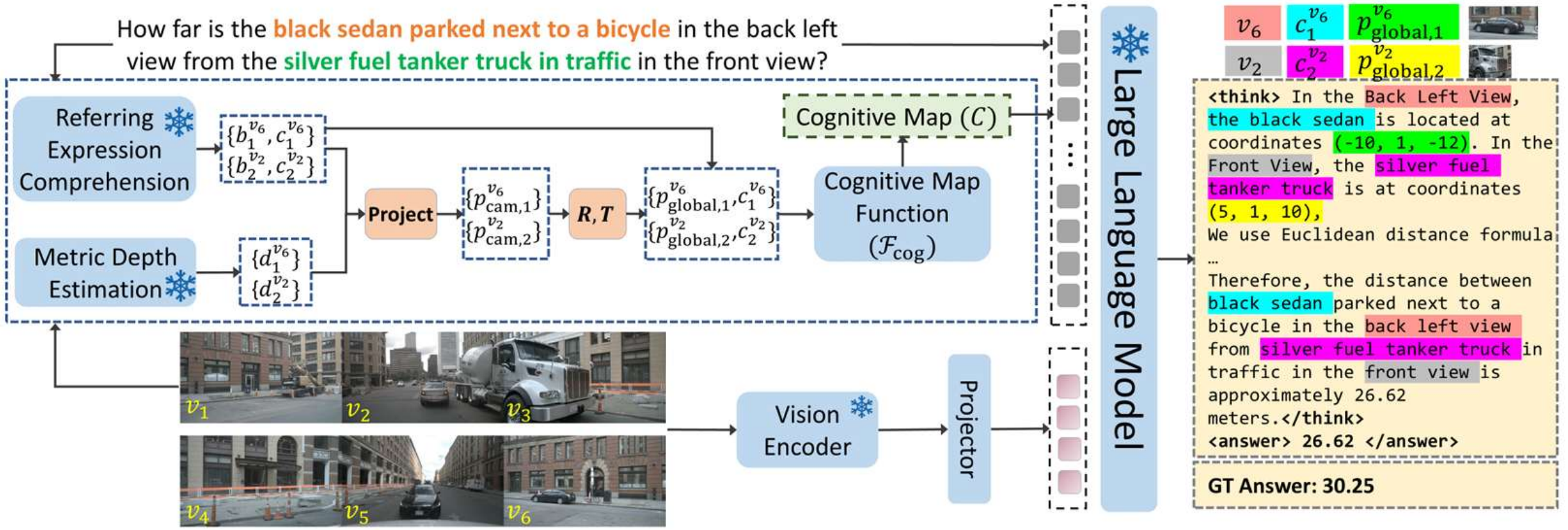
Motion Reasoning: What if we move to north?

Localization: Where is a specific object?



Spatial Reasoning with Vision-Language Models in Ego-Centric Multi-View Scenes (Arxiv 2511)

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing



DepthLM: Metric Depth From Vision Language Models (Arxiv 2510)

Principal axis distance: How far is this point from the camera in the forward backward direction?

- Ground Truth: 4.40m
- GPT-5: 2.7m
- **Ours: 4.30m**

Single Image Single Point

Speed: How many meters per second should we move in order to reach this point in exactly 4.0 seconds?

- Ground Truth: 1.76m/s
- GPT-5: 250m/s
- **Ours: The point is around 7.23 meters away. Hence, the speed should be around 7.23 / 4.0 = 1.81m/s**

Time: How many seconds do we need to reach this point if we move towards it with the speed of 6.0m/s?

- Ground Truth: 7.75s
- GPT-5: 2.5s
- **Ours: The point is around 48.28 meters away. Hence, we need around 48.28 / 6.0 = 8.05s**

(Single Image Single Point) Reasoning

Two point distance: How far are these 2 points from each other?

- Ground Truth: 2.75m
- GPT-5: 9.87m
- **Ours: 2.58m**

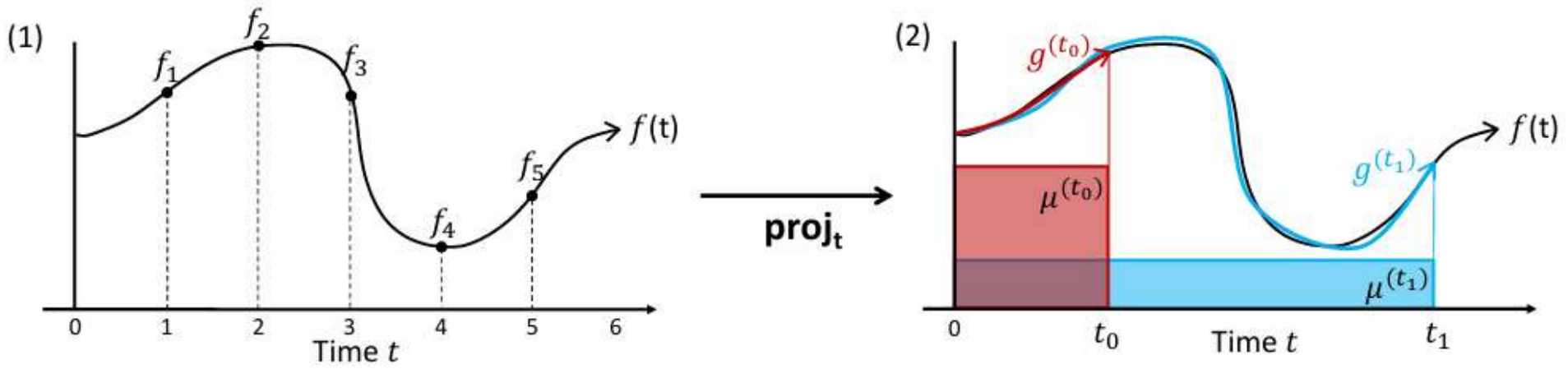
Multi-Point

Metric Scale Camera Pose: How many meters has the camera moved between these 2 images?

- Ground Truth: 5.94m
- GPT-5: 0m
- **Ours: 5.62m**

Multi-Image

HiPPO for VLN



Online Function Approximation

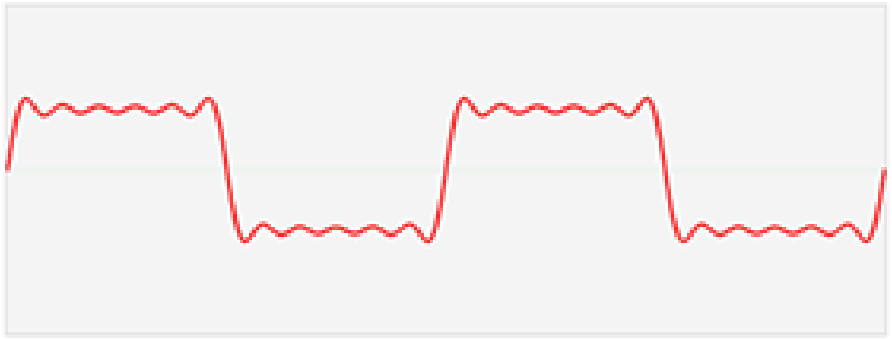
Gu A, Dao T, Ermon S, et al. Hippo: Recurrent memory with optimal polynomial projections, 2020.

Gu A, Goel K, Ré C. Efficiently modeling long sequences with structured state spaces, 2021.

HiPPO for VLN

A set of **complete polynomials** can approximate an arbitrary function by **linear combination**.

- Fourier series
- Legendre polynomials



Legendre polynomials：

$$P_n(x) = \frac{1}{2^n n!}\frac{d^n}{dx^n}[(x^2-1)^n]$$

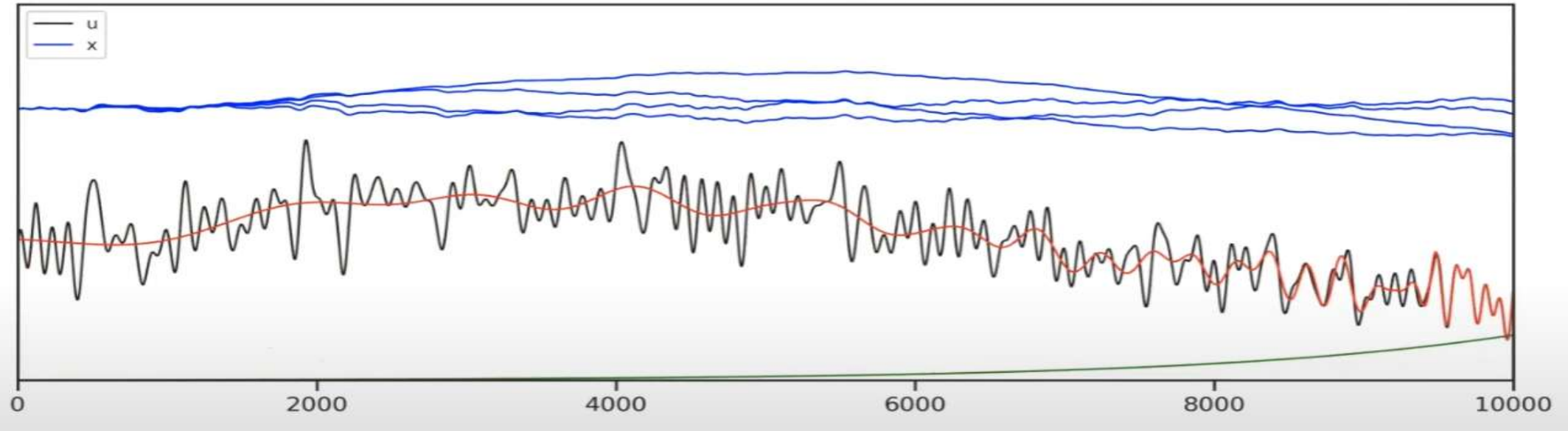
The Fourier transformation allows us to decompose a signal into sinusoidal functions

https://en.wikipedia.org/wiki/Fourier_transform

HiPPO for VLN

$$h_t = \overline{A}h_{t-1} + \overline{B}x_t$$
$$y_t = Ch_t$$
$$\overline{A} = \exp(\Delta A)$$
$$\overline{B} = (\Delta A)^{-1}(\exp(\Delta A) - I) \cdot \Delta B$$

A matrix can build in such a way that it approximates all the input signal seen so far into a vector of coefficients by Legendre polynomials.



把历史压缩问题形式化为针对随时间变化权重测度下的最佳多项式投影问题
利用正交多项式导出可在线更新的记忆策略