



Urbex: Agentic Spatial Grounding for City-Scale 3D Scenes

Yu Wang · Rui Dai · Wei Chen · Yi Wang

Bo Dang · Kaikui Liu · Xiangxiang Chu · Yansheng Li

Wuhan University · DreamX Team, Alibaba Group

NeurIPS 2026

-  1 Background and motivation
-  2 Datasets and method
-  3 Experiments and analysis
-  4 Conclusions and outlook

Background and motivation

Language-guided understanding of large urban scenes

Urban 3D maps contain rich geometry over large areas.

A language query must connect local appearance with distant spatial context.



The grounding challenge

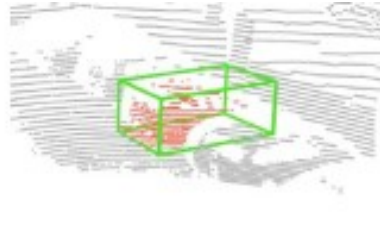
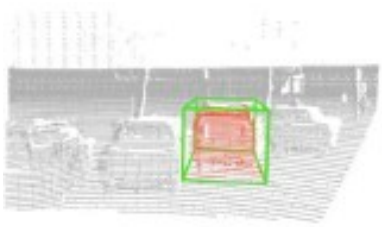
Find one target among many similar objects.

Preserve small-object details.

Use streets and named landmarks to identify the right region.

Background and motivation

3D grounding in urban environments



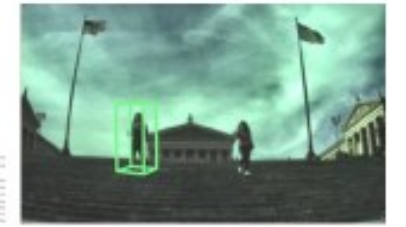
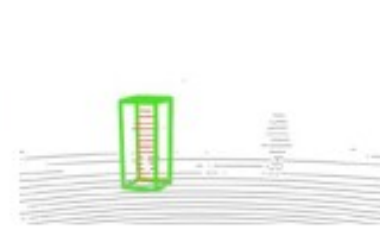
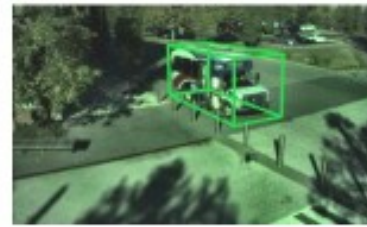
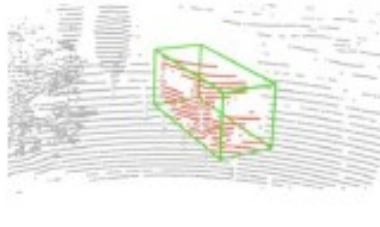
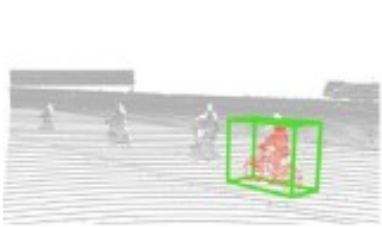
White utility truck parked on the side, with its rear facing the camera. It is the second car counting from right.



White car in the parking lot, viewed from an elevated angle, with three cars to its left and two to its right.



Pedestrian sitting on a ledge in the back, observing the surroundings in what appears to be a skate park.



Person riding a bicycle on the right side of the sidewalk, moving slowly, with three other riders in the front.



Stationary bus in rectangular shape is located in the center of the image, viewed from an elevated perspective.



Pedestrian standing still on the left side of the image, facing the camera, with another person on the right side.



Vehicle view

Aerial view

Robot view

- Applications include language-guided navigation and urban inspection

Background and motivation

Grounding at city scale: input and computation

Input scene and query

Query

The **small white building** with a flat roof situated just north of the Widney Alumni House, distinguishing it from the red-roofed buildings to the west and the large gray complex to the east.

A small target in a large spatial context



City-scale point cloud: 94.7 million points

Existing approaches



Only
50k points

DownSampling

Model

Result

Indoor 3D grounding [1]



Score: 0.98



Score: 0.93



Score: 0.88

Score every
candidate
(~1.3k)

Candidate ranking [2]

[1] Wu, Yanmin, et al. "Eda: Explicit text-decoupling and dense alignment for 3d visual grounding." *CVPR*. 2023.

[2] Li, Jinpeng, et al. "CityAnchor: City-scale 3D Visual Grounding with Multi-modality LLMs." *ICLR*. 2025.

Background and motivation

Active perception for large visual inputs

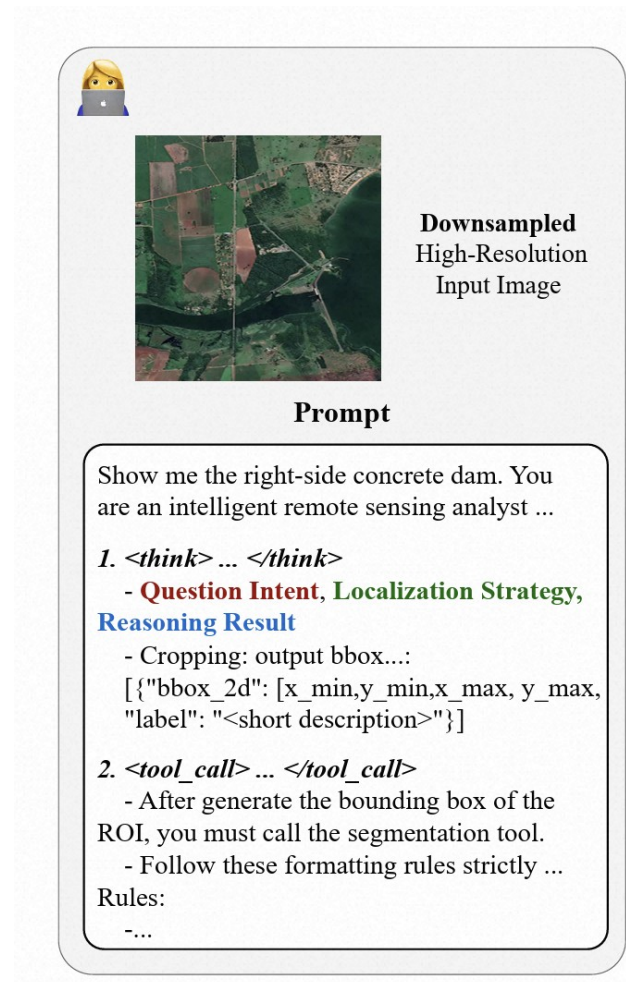
A fixed global view loses local detail

A search policy can choose which region to inspect next.

Zooming retrieves detail from the original high-resolution scene.

Multiple views provide complementary evidence before the agent commits a prediction.

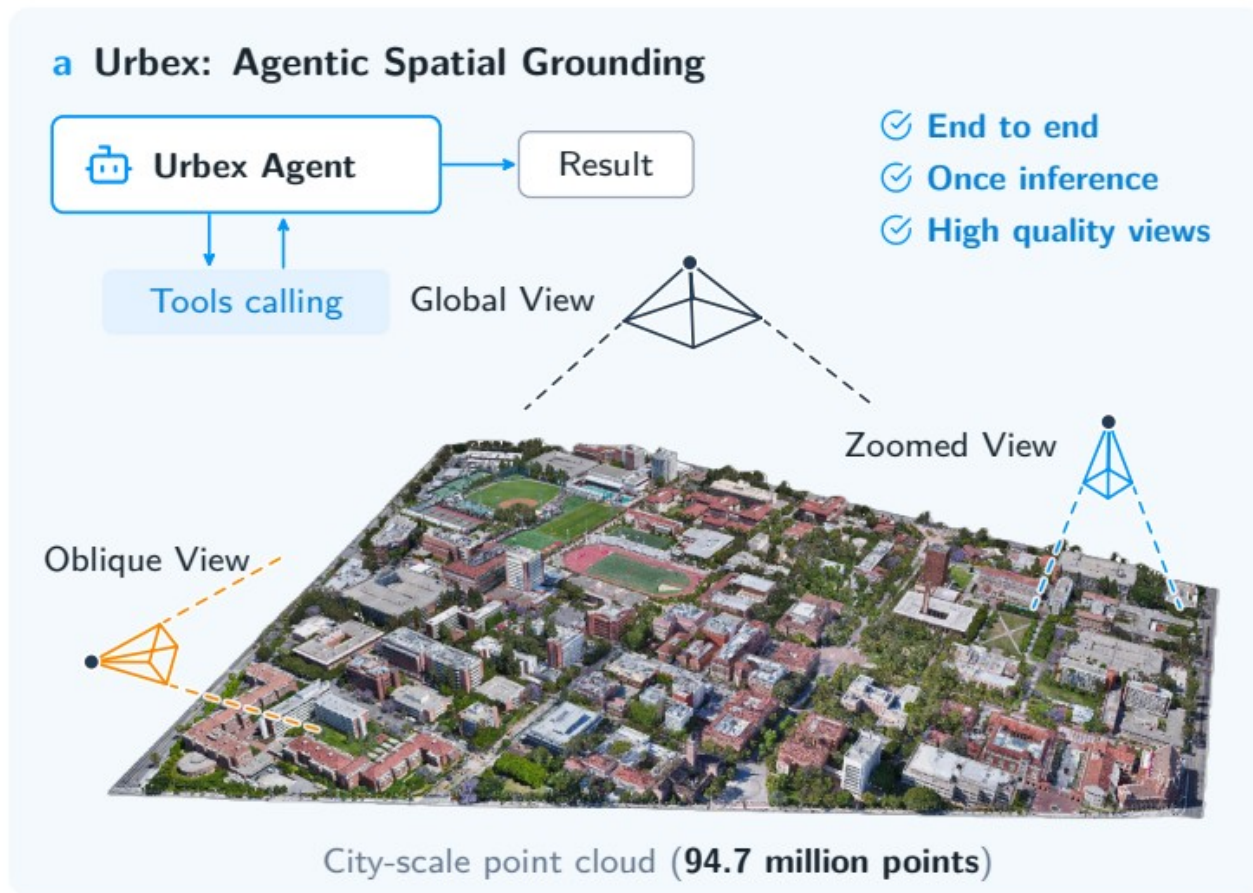
Acquire the visual evidence needed for the current decision



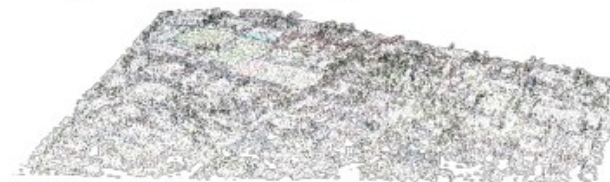
Tool-guided image inspection [1]

Background and motivation

Urbex formulates grounding as active spatial search



b Regular Grounding Method



Model input (**only 50k points**)

DownSampling → Model → Result

c Ranking-based Grounding Method



Score: 0.98



Score: 0.93



Score: 0.88

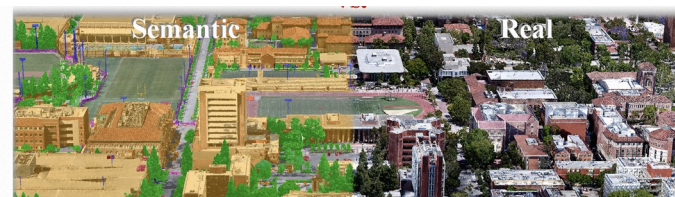
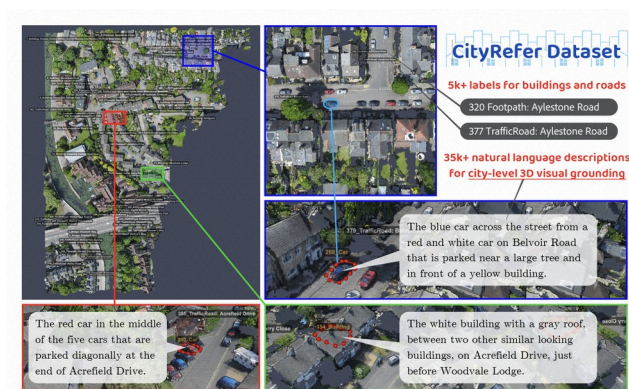
1.3k instances need scoring

Urbex predicts a free-form box. Candidate boxes are not inputs at inference.

Datasets and method

CityRefer training and STPLS3D-Refer evaluation

Dataset	Role	Data used
CityRefer	Training and in-domain tests	12 urban scenes
STPLS3D-Refer	Zero-shot transfer	445 expressions, 2 scenes



Benchmark construction

1. Extract and verify instances
2. Register landmarks
3. Generate descriptions
4. Filter and manually inspect

Datasets and method

An interactive multi-view environment and a learned policy



The environment maps a predicted box from a BEV-aligned view to world coordinates.

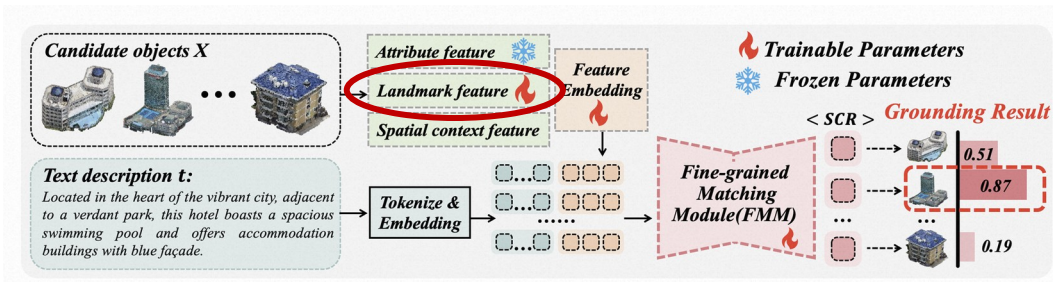
Datasets and method

City-aware tools

(a) There is a **two-section black building** that is one house to the left of the gray house at the intersection with **Brunswick Cottages**. There is a dark red car beside it.

(b) A **beige building** with a dark gray roof, that is attached to a large parking lot with four rows of cars, off the **Fitzroy Lane**, this building has three chimneys on the roof.

Landmarks in the query



Landmark features in prior work

Landmark
Query



Zoom-in



Side-view evidence
for verification

Oblique
Rendering

Tool interface

Tool	Input	Returned evidence
Landmark Query	Landmark name and source view	World position and projection into the requested view
Zoom-in	Source view and normalized crop box	High-resolution local BEV, new view ID, coordinate transform
Oblique Rendering	Candidate region and viewing direction	Oblique image for shape, height and spatial verification

Final answer: a box in a BEV-aligned view

The environment projects it to world coordinates. Oblique views support verification.

Source: final paper, Table 5.



Datasets and method

Observation, tool use and final prediction

Policy input

The referring expression

The current visual observations

Previous tool calls and feedback

View IDs and coordinate conventions

Policy output

A reasoning trace followed by a tool call,
or a final box in a known BEV view

Example query

“The blue car between an empty parking space and a red car in the parking lot at Buckingham Room.”

Search step

Resolve Buckingham Room in the global view, then inspect the nearby parking area at higher resolution.



Datasets and method

Reinforcement learning for active spatial search

Terminal
reward

Tool
reward

$$R_{\text{term}} = \lambda_{\text{iou}} \text{IoU}(\hat{b}, b^*) + \lambda_{\text{dist}} \exp\left(-\frac{d(\hat{b}, b^*)}{\tau}\right) - \lambda_{\text{step}} \frac{T}{T_{\text{max}}},$$

$$R(\tau) = R_{\text{term}} + \sum_{t=1}^T R_{\text{tool}}(s_t, a_t).$$

GRPO
training

$$A_i = \frac{R(\tau_i) - \mu_R}{\sigma_R + \epsilon},$$

$$\mathcal{J}_{\text{GRPO}}(\theta) = \frac{1}{G} \sum_{i=1}^G \frac{1}{T_i} \sum_{t=1}^{T_i} \left[\min(r_{i,t}(\theta) A_i, \text{clip}(r_{i,t}(\theta), 1 - \epsilon_c, 1 + \epsilon_c) A_i) - \beta D_{\text{KL}}^{i,t} \right].$$

Training configuration

Base model: Qwen3-VL-4B-Instruct

GRPO, without supervised trajectories

8 sampled trajectories per group

300 optimization steps

At most 8 assistant turns

The reward combines box overlap, center distance and tool-level shaping.



Experiments and analysis

Comparison methods

Prompted VLMs with the same tools

GPT-5.4 and Qwen3.6-plus use the same visual interface, action format and turn budget.

Indoor 3D grounding

EDA processes a downsampled 40k-point scene. InstanceRefer ranks provided instances.

City-scale candidate ranking

CityRefer and CityAnchor score candidate instances. We retrain both on the CityRefer split.

Experiments and analysis

Evaluation protocol

Split	Queries	Purpose
CityRefer train / validation	2,744 / 549	Policy learning / validation
CityRefer ND / NO	1,098 / 1,396	Novel descriptions / held-out scenes
STPLS3D-Refer	445	Zero-shot transfer

With GT candidates

Report Hit@1, Hit@3 and Hit@5.
Urbex predicts freely, then matches its box to candidates only for evaluation.

Without GT candidates

Report 2D IoU, 3D IoU and point F1.
Methods must output a spatial prediction without ground-truth boxes.

CityRefer results

Method	ND H@1	ND H@3	ND H@5	NO H@1	NO H@3	NO H@5	Time (s)
GPT-5.4	9.3	12.4	15.7	13.1	18.6	20.1	n/a
Qwen3.6-plus	6.4	9.6	11.9	11.5	16.2	16.8	n/a
InstanceRefer	1.6	5.0	10.5	4.9	15.9	24.5	0.6
EDA	0.1	0.1	0.2	0.2	0.3	0.4	3.0
CityRefer	1.6	3.9	4.8	0.6	2.1	3.4	0.7
CityAnchor	23.4	35.3	45.4	14.2	22.4	31.0	32.1
Urbex	25.8	39.2	42.3	23.9	36.9	40.7	3.2

Without ground-truth candidates

Method	ND 2D IoU	ND 3D IoU	ND F1	NO 2D IoU	NO 3D IoU	NO F1
GPT-5.4	4.4	2.2	9.4	4.6	1.9	10.4
Qwen3.6-plus	3.0	1.8	9.5	4.3	2.1	12.1
EDA	0.2	0.2	0.3	0.5	0.4	0.8
Urbex	15.6	11.0	20.1	14.6	7.7	17.8

Table 1. Main-run results. H@k = Hit@k. Online time includes model generation and tools, excluding offline preparation.

Experiments and analysis

Zero-shot transfer to STPLS3D-Refer

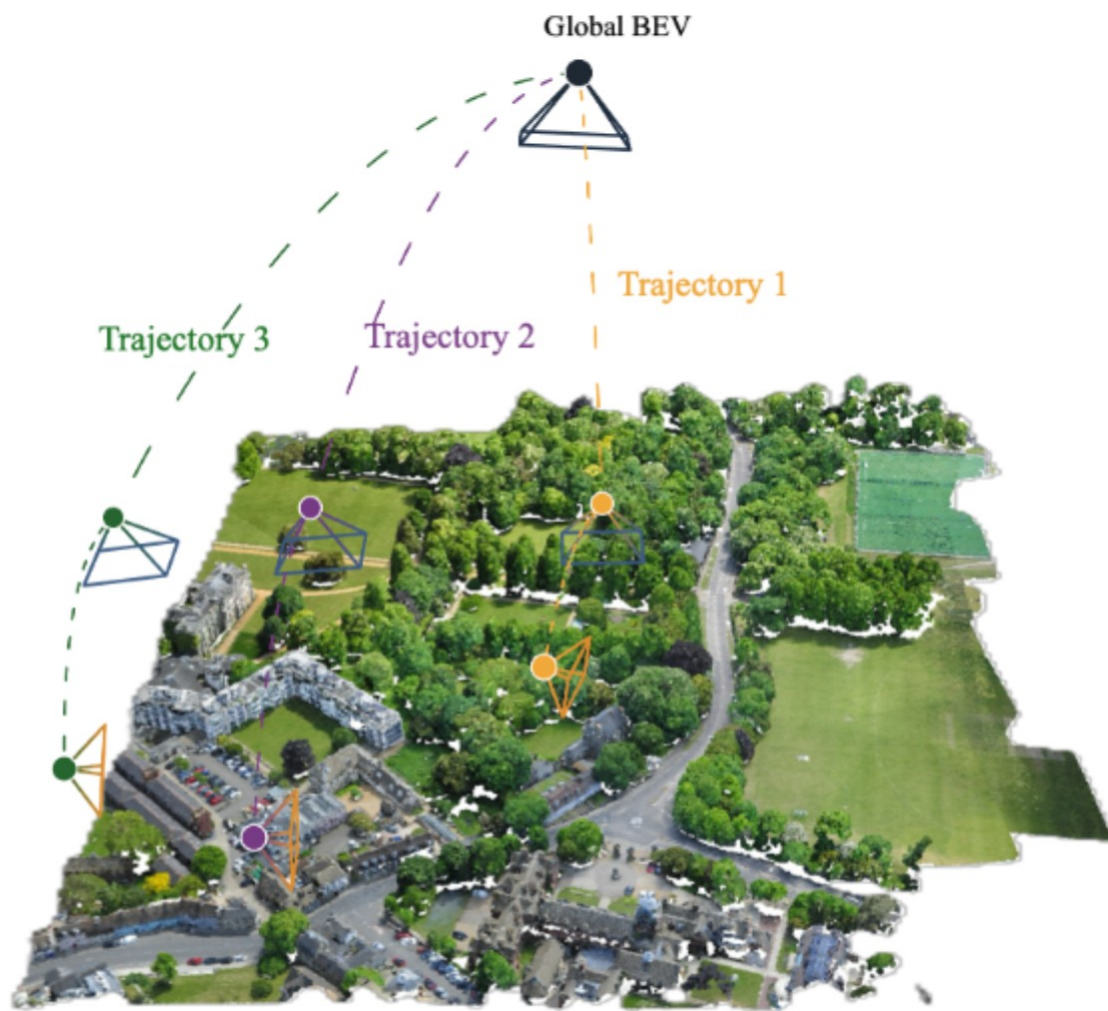
Method	Hit@1	Hit@3	Hit@5	2D IoU	3D IoU	Point F1	Time (s)
GPT-5.4	16.3	17.8	19.1	8.0	5.3	12.1	n/a
Qwen3.6-plus	12.0	13.8	14.3	8.1	6.4	12.2	n/a
InstanceRefer	3.8	7.8	12.3	n/a	n/a	n/a	0.6
EDA	0.0	0.0	0.5	0.4	0.2	0.4	2.9
CityRefer	1.4	2.5	3.8	n/a	n/a	n/a	1.4
CityAnchor	2.1	4.2	6.0	n/a	n/a	n/a	47.2
Urbex	27.2	32.1	33.4	13.0	10.0	16.2	4.8

The CityRefer-trained policy leads every reported accuracy metric on the new scenes.

Source: final paper, Table 2. No further training on STPLS3D-Refer. Metrics are percentages.

Experiments and analysis

Qualitative examples on CityRefer



The **white and gray building** on Queen's Road to the right of the Merton House.

'Merton House'
'Queen's Road'

Landmark Query



Zoom-in



Oblique Render (result)



GT

The **blue car** that is parked near the Art Room building.

'Art Room'

Landmark Query



Zoom-in



Oblique Render (result)



GT

The **blue car** between an empty parking space and a red car in the parking lot at Buckingham Room.

'Buckingham Room'

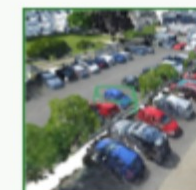
Landmark Query



Zoom-in



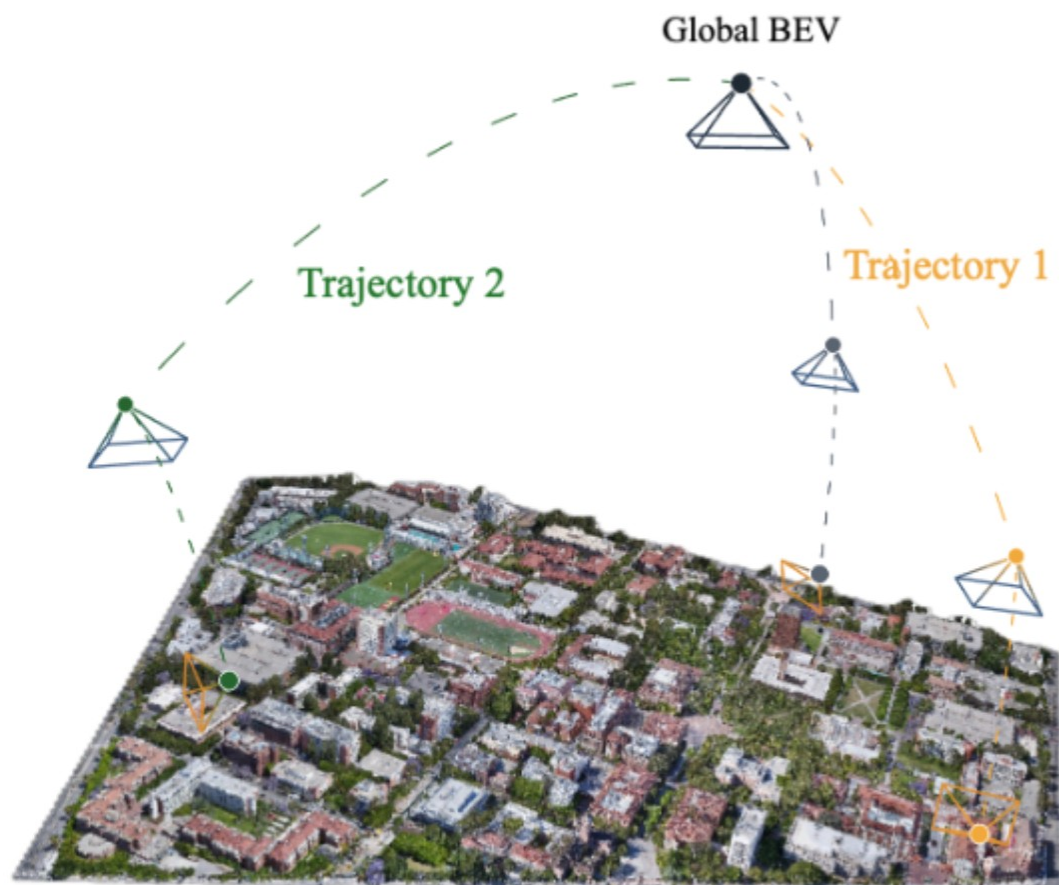
Oblique Render (result)



GT

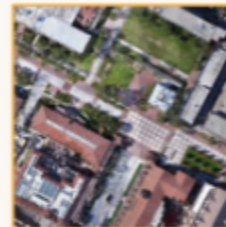
Experiments and analysis

Qualitative examples on STPLS3D-Refer



The **small white building** with a flat roof situated just north of the Widney Alumni House, distinguishing it from the red-roofed buildings to the west and the large gray complex to the east.

'Widney
Alumni House'



Zoom-in



Oblique Render (result)



GT

Landmark Query

The **red car** parked along the east side of South Vermont Avenue, situated north of two yellow cars and near the Scene Dock Theatre.

'South Vermont
Avenue'

'Scene Dock
Theatre'



Zoom-in



Oblique Render (result)



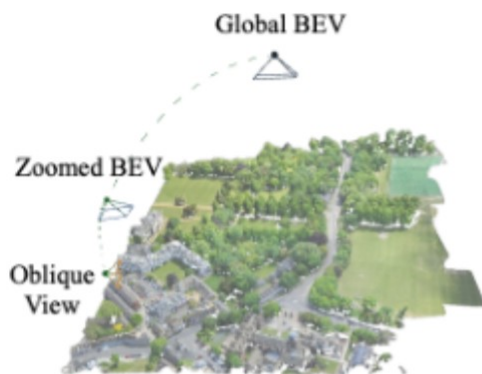
GT

Landmark Query

Experiments and analysis

A complete search trajectory

 The blue car between an empty parking space and a red car in the parking lot at Buckingham Room.



Step 1 · Landmark Query

look up a named landmark on the global BEV

THOUGHT

To locate the blue car between an empty parking space and a red car, I should first identify the Buckingham Room, as it is likely to be near the parking lot mentioned. Using the landmark lookup tool, I can find the coordinates of the Buckingham Room to narrow down the area where the blue car might be.

ACTION

```
landmark_query(names=[Buckingham  
Room] view_id="global")
```

RESULT

```
get_landmark_xy results:  
- 'Buckingham Room' → world  
(360.0,1528.3); on view global at  
(897,179)
```



Step 2 · Zoom In

agent crops a region of interest

THOUGHT

Now that I have the coordinates of the Buckingham Room at (897, 179) on the global view, I can focus on this area to locate the blue car between an empty parking space and a red car. The next step is to zoom in on this region to better visualize the parking lot and identify the specific car mentioned.

ACTION

```
zoom_in(view_id="global"  
bbox_2d=[800, 100, 1000, 300])
```

RESULT

```
zoom_in ok → new view 'Z1'
```



Step 3 · Oblique Render

verifies a candidate bbox in 3D - prediction (red)

THOUGHT

Now that I've zoomed in to view Z1, I can clearly see the parking lot near the Buckingham Room. The description mentions a blue car between an empty parking space and a red car. In the zoomed-in view, I can identify a row of parked cars, including red and blue vehicles. The blue car in question is likely to be adjacent to a red car and an empty space. I will now use the render_oblique tool to visually confirm the 3D structure of the blue car and its surroundings, ensuring it matches the description.

ACTION

```
oblique_render(view_id="Z1"  
bbox_2d=[515, 890, 565, 935]  
direction="east")
```

RESULT

```
render_oblique ok
```



Step 4 · Submit Answer

agent's final answer (no tool call)

THOUGHT

The 3D oblique view has successfully highlighted the blue car, which is positioned between an empty parking space and a red car, matching the query's description. The red 3D wireframe in the oblique view confirms the correct location of the blue car. Now, I will commit to the final answer using the specified format, as the visual confirmation has been achieved.

ACTION

```
answer = {"bbox_2d": [515, 890,  
565, 935], "view_id": "Z1"}
```

RESULT

```
2D IoU = 0.678 → ✓ success
```

Experiments and analysis

Tool and reward ablations

Zoom	Landmark	Oblique	Turns	Hit@3	2D IoU
No	No	No	0	10.5	2.1
Yes	No	No	8	15.5	4.1
Yes	Yes	No	8	28.2	8.9
Yes	Yes	Yes	7	34.8	11.0
Yes	Yes	Yes	8	36.9	14.6
Yes	Yes	Yes	9	36.8	13.2

Reward	Hit@3	2D IoU
IoU only	22.8	6.7
IoU + distance	30.6	10.9
IoU + tools	28.4	9.6
Full reward	36.9	14.6

All tools contribute. Eight turns perform best in this budget comparison.
Distance and tool-level shaping complement the terminal IoU reward.

Source: final paper, Tables 3–4. CityRefer-NO. Metrics are percentages.

Experiments and analysis

Repeated decoding and learned execution

Dataset	Urbex Hit@1, mean $\pm$ SD	Best comparison
CityRefer-ND	25.62 $\pm$ 0.33	23.4, CityAnchor
CityRefer-NO	23.53 $\pm$ 0.73	14.2, CityAnchor
STPLS3D-Refer	26.34 $\pm$ 0.58	16.3, GPT-5.4

Five decoding runs of one checkpoint. These results measure inference sampling variation, not variation across training seeds.

Fixed tool-order diagnostic

Under the same fixed tool sequence, Hit@1 rises from 2.7 for the base model to 10.0 after RL. Replaying the tool order alone does not recover the full learned policy.



Conclusions and outlook

Inference efficiency and preparation cost

Component	Time
Urbex with cached tools	3.2 s/query
CityAnchor online inference	32.1 s/query
Urbex with on-demand oblique views	≈4.4 s/query (estimated)
Optional oblique cache for 12 scenes	≈2.3 hours, one time

**The warm online comparison is approximately 10× faster.
On-demand rendering retains an estimated 7.3× speedup.**

Table 17. Both online timings exclude offline preparation. Cache construction is not the full deployment cost.



Conclusions and outlook

Conclusions and limitations

Main conclusions

City-scale grounding can use active spatial search with no candidate boxes at inference.

The learned tool-use policy improves target identification and box overlap.

The same policy transfers to new urban scenes without further training.

Current limitations

Calibration and rendering errors can affect world-coordinate predictions.

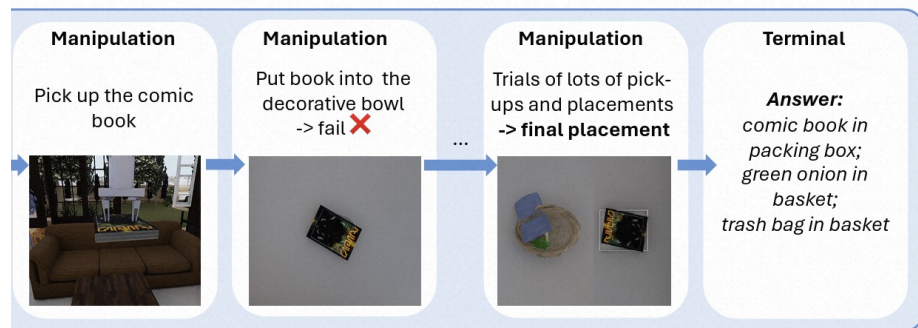
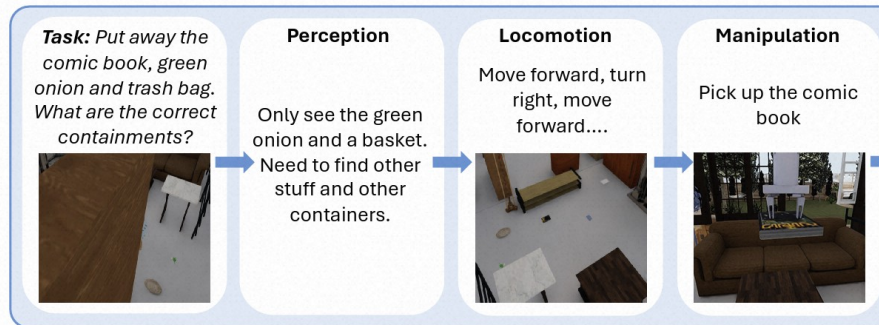
Missing or unreliable landmarks make search harder.

BEV-aligned boxes leave room for finer 3D segmentation and boundary refinement.

Conclusions and outlook

Future work: active perception in embodied environments

Future direction: extend language-guided visual search to sequential navigation and interaction. This requires new environments and evaluation beyond the grounding benchmarks in this work.



Example sequence

Action	Description
<i>Locomotion</i> move_forward / move_backward move_left / move_right move_up / move_down	Translate agent along viewing axis Translate agent laterally Translate agent vertically
<i>Perception</i> turn_left / turn_right turn_up / turn_down	Rotate agent horizontally Rotate agent vertically
<i>Manipulation</i> pick_up obj put obj inside obj put obj on obj fill obj with water pour obj from obj to obj	Pick up object Put object inside object Put object on object Fill object with water Pour object from object to object
<i>Terminal</i> answer($\hat{y}$, c)	Commit to final answer with confidence c

Possible action interface