# AerialVLN : Vision-and-Language Navigation for UAVs

(2023 ICCV)

# Content

- Introduction
- Motivation
- Method
- Exprtiment
- Results
- Conclusion

# Introduction

- UAV-based applications: goods delivery, traffic/security patrol,  scenery tour

- Vision-and-Language Navigation (VLN) tasks have drawn significant attention

# Motivation

- Research Gaps
  - Existing VLN tasks are for navigation on the ground (indoors /  outdoors)
  - Navigating in the sky is more complicated than on the ground
    - Larger action space
    - Complex spatial relationship reasoning
    - Longer path
    - Avoid getting stuck on objects in 3D space

- Propose AerialVLN:  UAV-based and towards outdoor environments

# Method

- Task
  - Input: front view perceptions (RGB & depth images)
  - Output: the agent predict a series of actions

- Dataset Creation
  - 3D simulator
  - 25 city-level scenarios
  - Supports continuous navigation

- Model

# Method - Dataset Collection

Two Steps

- Path Generation
  - Human manipulators: pass several random-selected landmarks from a pre-defined landmarker set
  - In simulator, provide hints about directions & distances to the next landmark to the manipulators
  - Output: multirotor’s pose trace
  - Remove redundant motion for smoother ground truth trajectories
- Instruction collection
  - Show videos of drone flight and require annotators to give natural language commands that can lead a pilot to complete the flying

# Method - Dataset Analysis

Comparision: AerialVLN dataset vs other popular VLN datasets

- Largest average path length
- Most average actions per path
- Number of instructions

Dataset Split

train, val_seen, val_unseen, test (unseen)

# Method - Model

Five baseline models are evaluated on the task

- Random action - reflect the size of solution space
- Action sampling according to the action distribution of the training set - measure the similarity of the action distribution on evaluation and training splits
- LingUNet
- Sequence-to-Sequence - ResNet (img) + LSTM(instruction) + GRU + Linear
- Cross-Modal Attention

# Method - Model (Cross-Modal Attention (CMA))

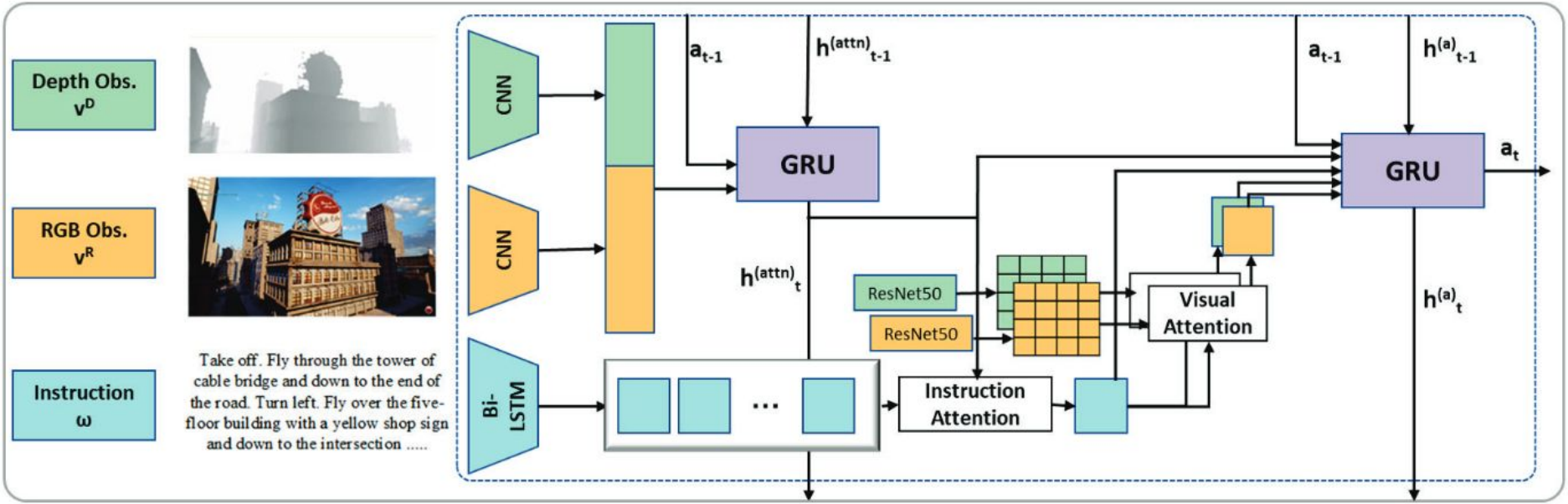



Figure 4: Main architecture of the Cross-Modal Attention model

# Method - Model (Look-ahead Guidance (LAG))

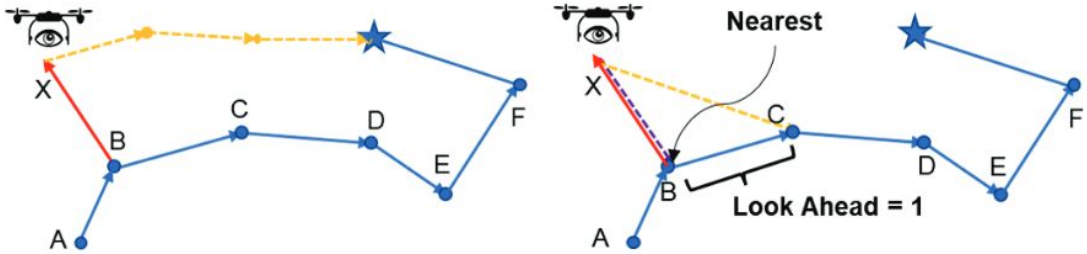
Motivation: in student-forcing fashion, ground-truth actions are usually determined via the shortest path from the current location to the destination

However, instructions do not describe the shortest path from the starting to the destination

# Method - Model（Look-ahead Guidance (LAG)）

3 Steps

1. Find the shortest path to return to the ground-truth (point B)
2. Navigate along the ground-truth path 10 steps (reach C)
3. The look-ahead path is the shortest path from X to location C, and the ground-truth action for the nexxt step is the first step on this path




(a) Shortest path guidance (b) Look-ahead guidance

‘A’ denotes starting location; ‘’ denotes destination; ‘X’ denotes current location; Blue path denotes ground-truth; Yellow path denotes “generated ground-truth” when the agent deviates from the real ground-truth path.

# Experiment - Metrics

- Success Rate - agent stops within 20m of the destination
- Oracle Success Rate - distance between the destination and any point on the trajectory < 20 m
- Navigation Error - distance between stop location to the destination
- Success rate weighted by Normalised Dynamic Time Wraping - considers both the navigation success rate the similarity between ground truth path and the model predicted path

# Results

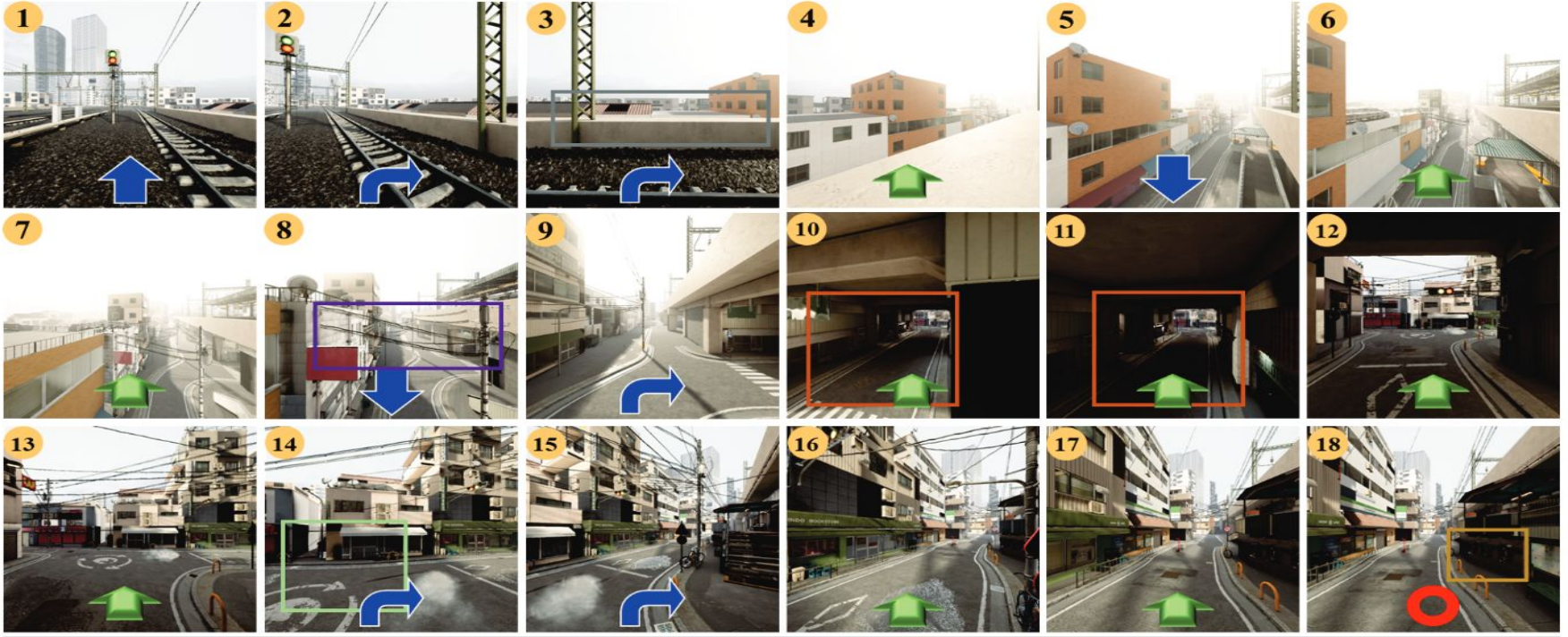
| # | AerialVLN | Validation Seen | | | | Validation Unseen | | | | Test Unseen | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/% ↑ | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/% ↑ | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/%↑ |
| 1 | Random | 300.8 | 0.0 | 0.0 | 0.0 | 351.0 | 0.0 | 0.0 | 0.0 | 356.3 | 0.0 | 0.0 | 0.0 |
| 2 | Action Sampling | 383.1 | 0.1 | 2.1 | 0.1 | 434.9 | 0.2 | 2.1 | 0.1 | 441.9 | 0.2 | 1.8 | 0.1 |
| 3 | Seq2Seq | 480.4 | 2.9 | 10.2 | 1.0 | 551.8 | 1.1 | 5.6 | 0.3 | 558.8 | 1.0 | 4.9 | 0.3 |
| 4 | CMA | 293.5 | 2.3 | 6.5 | 0.8 | 360.7 | 1.6 | 4.4 | 0.5 | 358.6 | 1.6 | 4.1 | 0.5 |
| 5 | Human | - | - | - | - | - | - | - | - | 73.5 | 80.8 | 80.8 | 14.2 |

| # | AerialVLN-S | Validation Seen | | | | Validation Unseen | | | | Test Unseen | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/% ↑ | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/% ↑ | NE/m ↓ | SR/% ↑ | OSR/% ↑ | SDTW/% ↑ |
| S1 | Random | 109.6 | 0.0 | 0.0 | 0.0 | 149.7 | 0.0 | 0.0 | 0.0 | 148.5 | 0.0 | 0.0 | 0.0 |
| S2 | Action Sampling | 213.8 | 0.9 | 5.7 | 0.3 | 237.6 | 0.2 | 1.1 | 0.1 | 242.0 | 0.7 | 2.5 | 0.3 |
| S3 | LingUNet | 383.8 | 0.6 | 6.9 | 0.2 | 368.4 | 0.4 | 3.6 | 0.9 | 399.8 | 0.1 | 3.1 | 0.1 |
| S4 | Seq2Seq | 146.0 | 4.8 | 19.8 | 1.6 | 218.9 | 2.3 | 11.7 | 0.7 | 214.6 | 2.2 | 9.4 | 0.7 |
| S5 | CMA | 121.0 | 3.0 | 23.2 | 0.6 | 172.1 | 3.2 | 16.0 | 1.1 | 178.5 | 3.9 | 13.1 | 1.4 |
| S6 | Seq2Seq-DA | 85.5 | 9.9 | 24.1 | 4.5 | 143.5 | 4.0 | 10.9 | 0.7 | 140.2 | 3.5 | 9.5 | 0.6 |
| S7 | CMA-DA | 92.2 | 9.9 | 26.5 | 3.7 | 122.7 | 4.5 | 13.9 | 1.0 | 125.4 | 4.3 | 14.8 | 1.2 |
| S8 | Ours (LAG) | 90.2 | 7.2 | 15.7 | 2.4 | 127.9 | 5.1 | 10.5 | 1.4 | 128.3 | 4.5 | 11.6 | 1.3 |

Table 4: Performance of baselines on our AerialVLN task (Row 1-5) and AerialVLN-S task (Row S1-S7). There is a significant gap to human performance.

# Results

**Instruction**: *Ascend [1] into the air then turn right [2] above the railway. Turn right [3] parallel with the curb. Descend [5] towards the electrical post at the intersection of the road. Turning right [9] down into the dark alleyway. Fly to [11] the end of the dark alleyway then turning right [14] at the intersection. Stop [18] near the payphones on the right side of the road.*



# Conclusion

- Introduce the AerialVLN task
- Approach to create & analyse datasets
- Model architecture