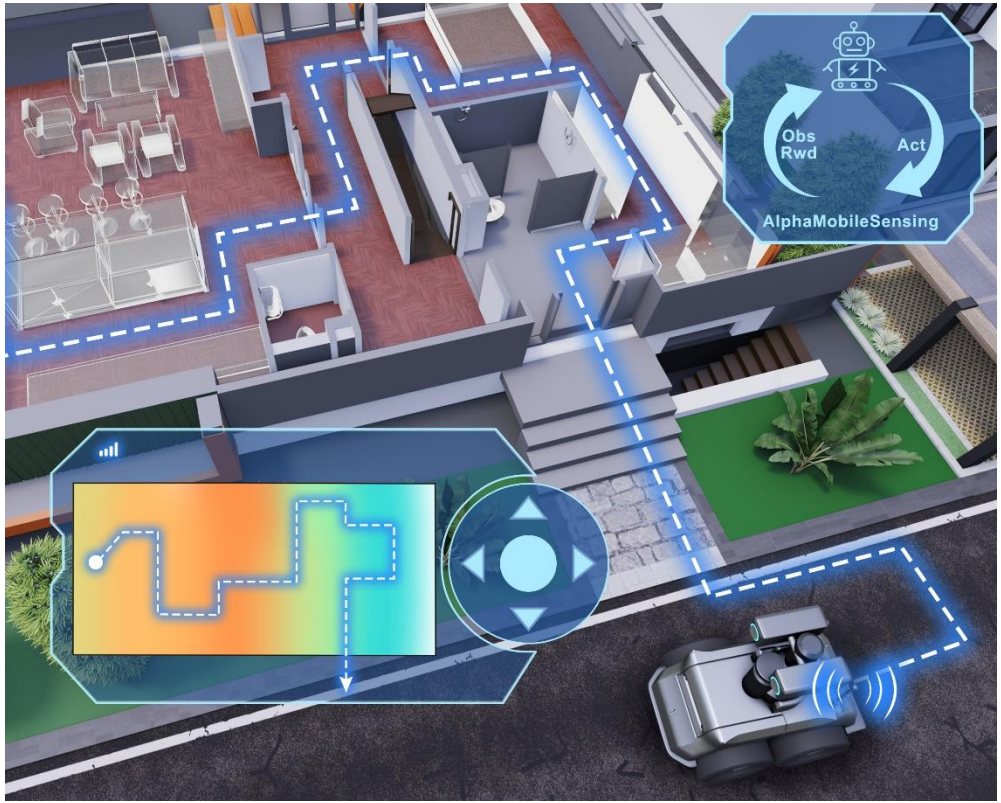
Agenda

- What is ***AlphaMobileSensing*** ?
- Why we need ***AlphaMobileSensing*** ?
- How to use ***AlphaMobileSensing*** ?
- How to learn ***AlphaMobileSensing*** ?

• What is ***AlphaMobileSensing*** ?

◦ A virtual testbed for developing, testing, benchmarking algorithms for mobile environmental monitoring

◦ Enables carrying out mobile environmental monitoring-related experiments in a virtual world



• Why we need ***AlphaMobileSensing*** ?

• Why we need environmental monitoring ?

◦ IEQ is important
◦ Prerequisite of environmental control

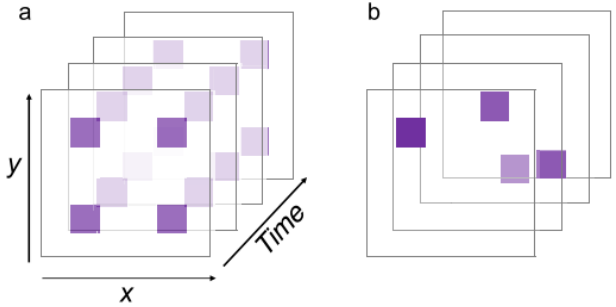
• Why we need mobile environmental monitoring ?

Stationary VS Mobile

Granularity
Deployment
Maintenance
Cost

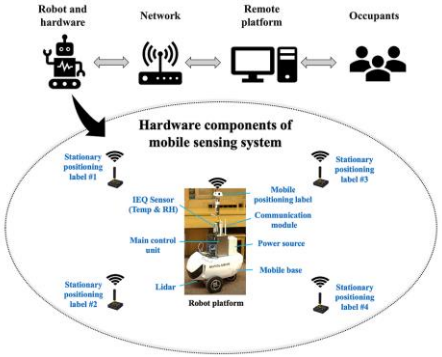
• Why we need ***AlphaMobileSensing*** ?

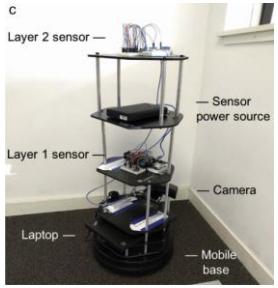
- What is the difficulty in mobile environmental monitoring ?

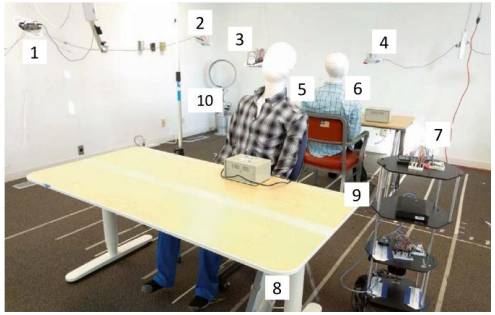


◦ Monitoring data is sparse in time and space (algorithm for field reconstruction)
◦ Need to instruct the robot where to sense (algorithm for path planning)
◦ Sophisticated algorithms are required

- What are the challenges in real-world experiment ?



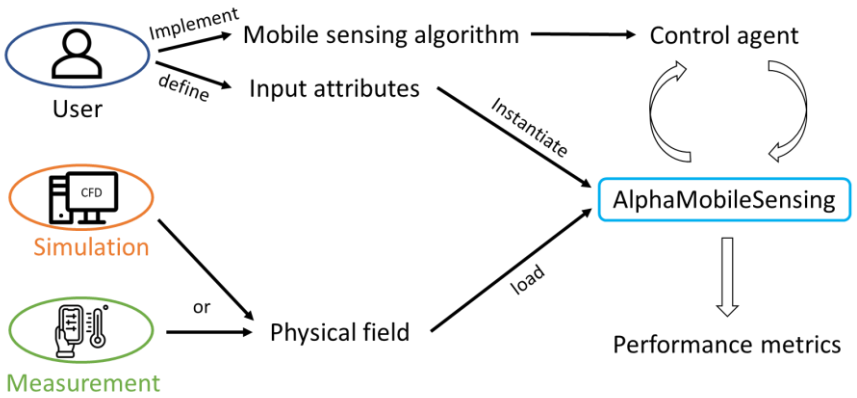




◦ Hardware
◦ Software
◦ Test site
◦ Accident

• How to use ***AlphaMobileSensing*** ?

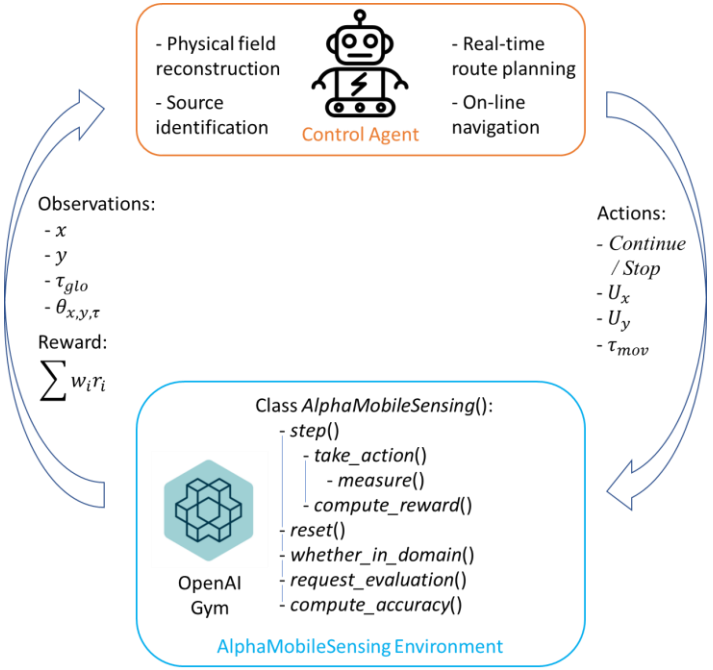
- How does it work ?



◦ Implement mobile sensing algorithm (Determine research object)

◦ Define input attributes (Set experiment configurations)

◦ Load physical field (Prepare virtual test site)

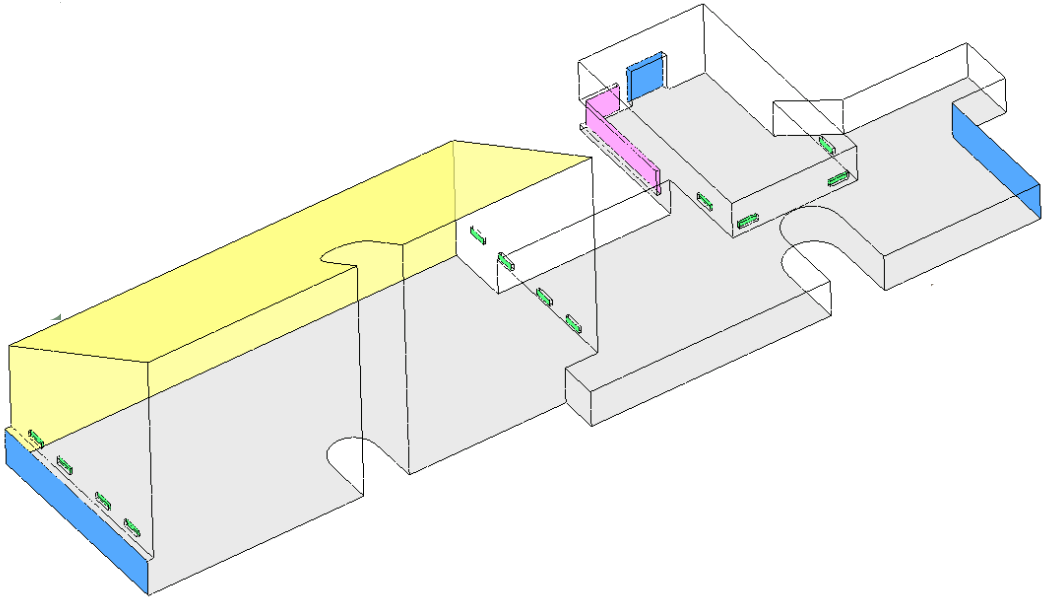


◦ Control agent interacts with the test site

• How to use ***AlphaMobileSensing*** ?

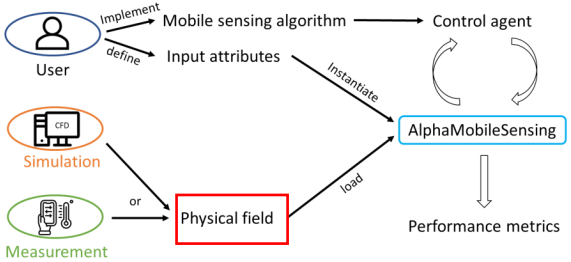
- An example for demonstration
  - Robot-based temperature monitoring in Chia-Wei Woo Academic Concourse

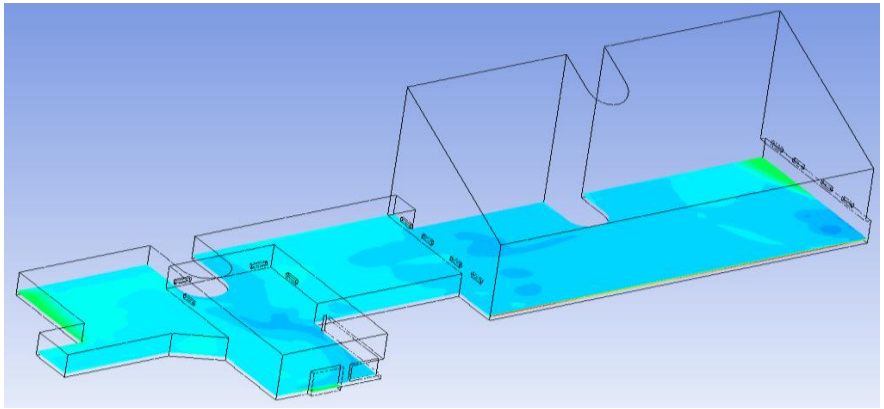
Environmental monitoring based on pre-defined path
Infer temperature distribution based on monitoring data



• How to use ***AlphaMobileSensing*** ?

- What preparations do we need ?
  - Dynamic physical field





Spatial coordinate

Time stamp

CSV

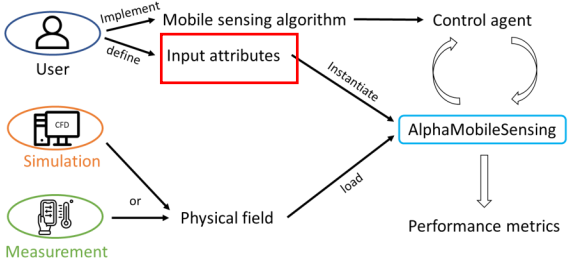
	A	B	C	D	E	F	G	H	I	J	K	L
1	X	Y	Z	0	1	2	3	4	5	6	7	8
2	-0.5	0	0.5	300	300.0005	300.0009	300.0007	300.0005	300.0003	300.0003	300.0002	300.0002
3	-0.5	0.096677	0.5	300	300	300	300	300	300	300	300	300
4	-0.5	0.213652	0.5	300	300	300	300	300	300	300	300	300
5	-0.5	0.274038	0.5	300	300	300	300	300	300	300	300	300
6	-0.5	0.35758	0.5	300	300	300	300	300	300	300	300	300
7	-0.5	0.665204	0.5	300	300	300	300	300	300	300	300	300
8	-0.5	1.299247	0.5	300	300	300	300	300	300	300	300	300
9	-0.5	1.330876	0.5	300	300	300	300	300	300	300	300	300
10	-0.5	1.894201	0.5	300	300	300	300	300	300	300	300	300
11	-0.5	1.966074	0.5	300	300	300	300	300	300	300	300	300
12	-0.5	2.543164	0.5	300	300.0001	300	300	300	300	300	300	300
13	-0.5	3.153794	0.5	300	300	300	300	300	300	300	300	300

Temperature

```
datapath='D:/Research Working Fol
der/Mobile Sensing Virtual Environ
ment/dynamic_50cm_DEMO.csv'
```

• How to use ***AlphaMobileSensing*** ?

- What preparations do we need ?
  - Input attributes

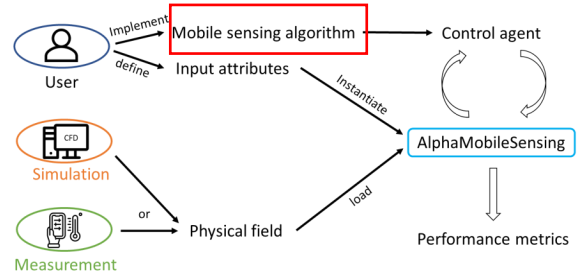


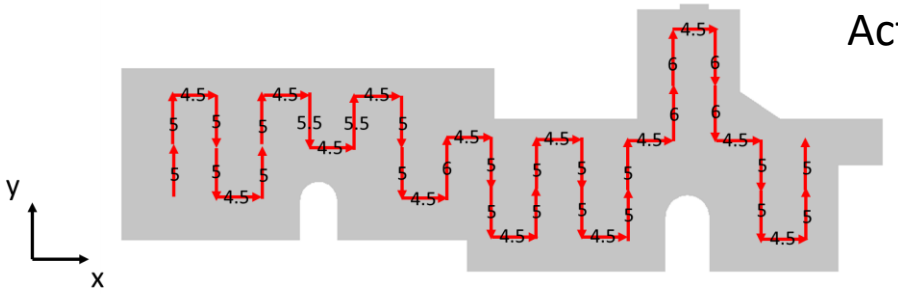

```
- PFdataPath: string, path of physical field data
- PFTHorizon: int, time horizon of a physical field
- PFTStepsize: int, time step size of a physical field
- AgentNumber: int, number of robots utilized in mobile sensing
- MeaDuration: int, time required by a robot to measure physical variables at a location
- IniLocation: tuple, initial location of a robot
- MaxSpeed: float, maximum moving speed of each robot
- CostWeight: tuple, weight between moving distance and moving time to compute reward
- MaxStep: int, maximum number of steps for an episode
```

```
from env import environment_v2
datapath='D:/Research Working Folder/Mobile Sensing Virtual Environment/dynamic_50cm_DEMO.csv'
env=environment_v2.AlphaMoSeEnv(datapath, 1200, 1, 1, (10, ), ((2.75, 3.0), ), 2, (0.5, 0.5), 1000)
```

• How to use ***AlphaMobileSensing*** ?

- What preparations do we need ?
  - Path planning algorithm (pre-defined trajectory)
  - Spatio-temporal interpolation algorithm





Action series

Moving velocity | Moving time | Moving Signal

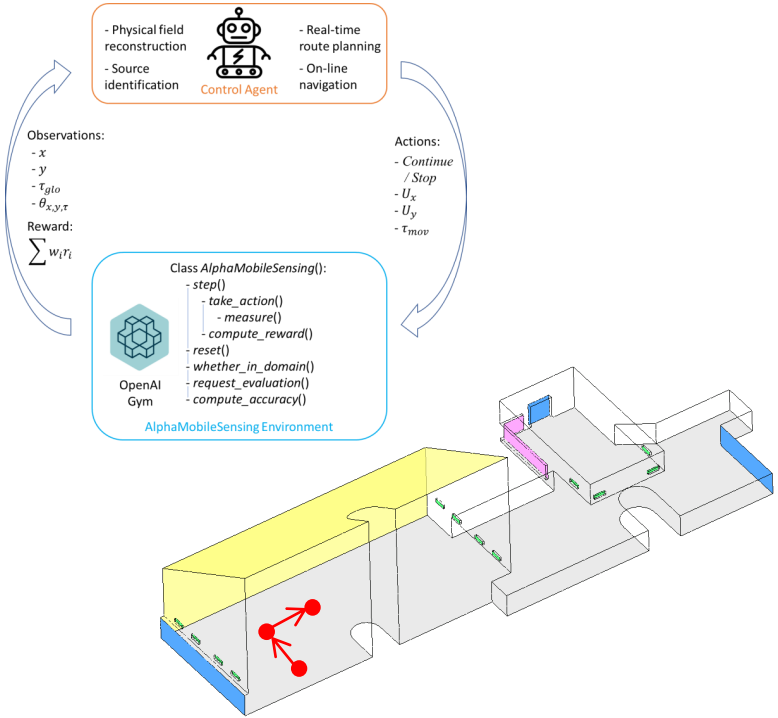
	A	B	C	D
1	Ux	Uy	tau	Continue/Stop
	0	0.2	25	1
	0	0.2	25	1
	0.1	0	45	1
	0	-0.2	25	1
	0	-0.2	25	1
7	0.1	0	45	1
8	0	0.2	25	1
9	0	0.2	25	1
10	0.1	0	45	1
11	0	-0.1	55	1
12	0.1	0	45	1
13	0	0.1	55	1
14	0.1	0	45	1
15	0	-0.2	25	1

```
action_series_df=pd.read_excel('D:/Research Working Folder/Mobile Sensing Algorithm Testing/Code/data/action_series.xlsx')
```

• How to use ***AlphaMobileSensing*** ?

- How does it work in an experiment

moving ⟶ monitoring ⟶ moving ⟶ …




```
env.reset()

num_agent=1

measurement_log_raw=np.full((len(action_series_df)+1,4*num_agent), np.nan)

for i in range(num_agent):
    measurement_log_raw[0,4*i:4*(i+1)]=env.obs[i]


for i in range(len(action_series_df)):

    action=[]
    for j in range(num_agent):
        array=action_series_df.iloc[i,4*j:4*(j+1)].values
        action.append(array)

    step_log=env.step(action)


    if step_log[0]==True:
        break
    else:
        for j in range(num_agent):
            measurement_log_raw[i+1,4*j:4*(j+1)]=env.obs[j]
```

Initialize an experiment

Get the 1st measurement at the initial state

Moving to and get measurement at a new position

• How to use ***AlphaMobileSensing*** ?

- We can obtain a monitoring log as the experiment ended

Spatial coordinate

Time stamp

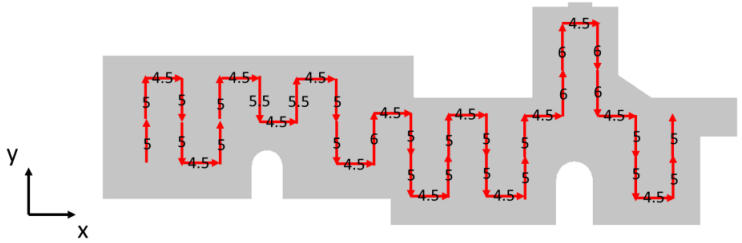
Measurement

	A	B	C	D	E
1		0	1	2	3
2	0	0.25	0.5	0	297
3	1	0.25	3	15	295.3861
4	2	0.25	5.5	30	296.7384
5	3	2.75	5.5	45	296.6899
6	4	2.75	3	60	296.7988
7	5	5.25	3	75	296.6418
8	6	5.25	5.5	90	296.6177
9	7	7.75	5.5	105	296.5107
10	8	7.75	3	120	296.4615
11	9	7.75	0.5	135	296.8434
12	10	5.25	0.5	150	296.2414
13	11	2.75	0.5	165	296.1713

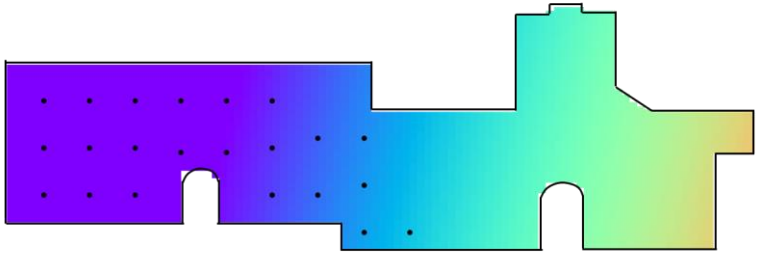
```
ml=pd.DataFrame(measurement_log_raw)
ml.to_csv('D:/Research Working Folder/Mobile Sensing Algorithm Testing/Scenario D/measurement_log.csv')
```

• How to use ***AlphaMobileSensing*** ?

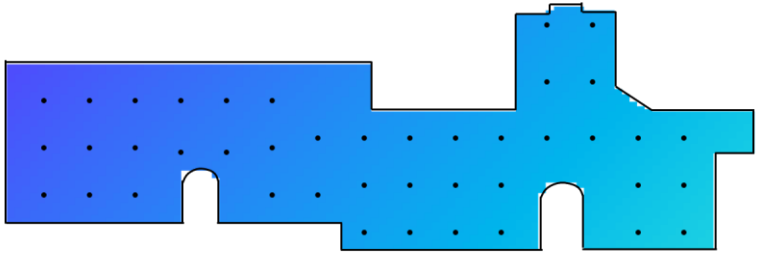
• Infer temperature distribution based on monitoring data via a field reconstruction algorithm (spatial-temporal interpolation)

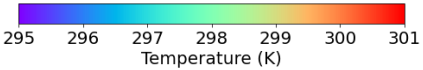


Spatial-temporal interpolation (1065 s)



Spatial-temporal interpolation (2005 s)





• How to use ***AlphaMobileSensing*** ?

- How to evaluate the algorithm ?
  - Compare inferred results with the ground truth

User can determine how many points and at which time point for accuracy evaluation

```
env.request_evaluation()
```

'Please input sampling number:'

'Please input target time:'

'Please input a path for template export:'

Quantify the discrepancy via RMSE or MAE

Root mean square error (RMSE)
Mean absolute error (MAE)

$$RMSE = \sqrt{\frac{\sum (y_i - y_p)^2}{n}}$$

$$MAE = \frac{|(y_i - y_p)|}{n}$$

```
env.compute_accuracy()
```

Spatial coordinate

Time stamp

CSV

	A			
1	Unnamed	X	Y	1065
	12209	56.73621	21.92175	
	2364	32.6409	2.297076	
	11358	54.90479	22.95	
	11921	56.49178	22.88687	
	4899	49.98569	15.95021	
7	7526	50.42731	18.47517	
8	2982	39.05226	1.218835	
9	2690	35.49131	16.15081	
10	8750	51.22775	21.81089	
11	2145	28.71453	1.812448	
12	5229	49.99915	14.81835	
13	11000	54.04471	22.72923	
14	3815	49.82527	13.4177	
15	918	9.641085	1.765702	
16	8866	51.35503	22.0097	
17	9629	52.24178	12.5472	
18	10579	53.62172	22.29784	

Inferred results inputted by users

• How to learn ***AlphaMobileSensing*** ?

◦ Official repository:
https://github.com/kishuqizhou/AlphaMobileSensing

Where you can find
- Source code of **AlphaMobileSensing** (with annotations)
- Demo files for reproducing the work in the paper

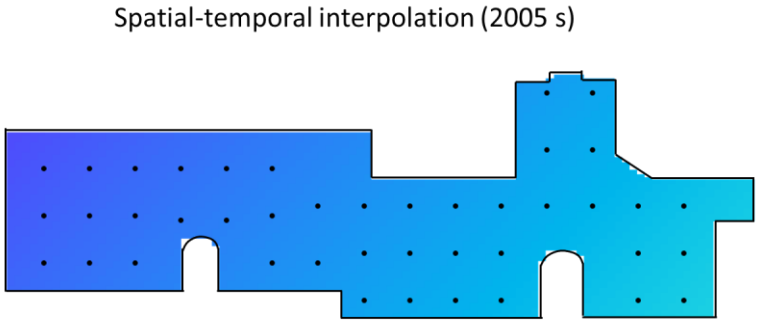
Feel free to play with ***AlphaMobileSensing*** !

• How to learn ***AlphaMobileSensing*** ?

Homework

◦ Please refer to the Demo file to get monitoring log at 2005s

◦ Please refer to the Demo file to use the spatio-temporal interpolation algorithm to infer the temperature distribution at 2005s



https://github.com/kishuqizhou/AlphaMobileSensing/blob/main/demo/Demo_Dynamic_v2.ipynb