Location/Area Estimation Based on Multi-Class Classification with DNN Using RSSI of Wireless LAN

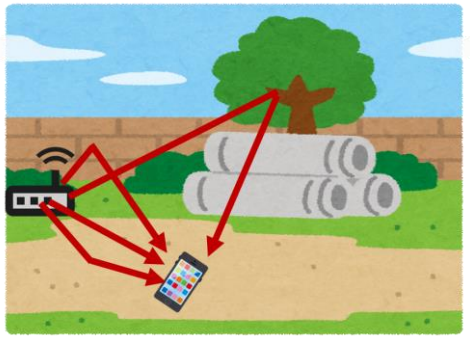
Team: DLine

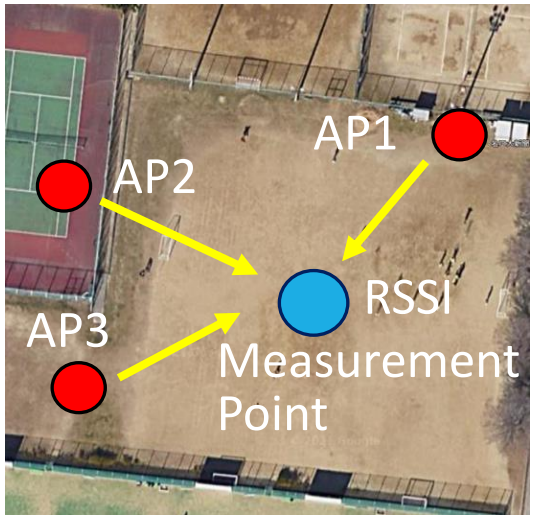
Shota Miyagoshi (University of Fukui)

Tomonori Yokono (University of Fukui)

Challenge

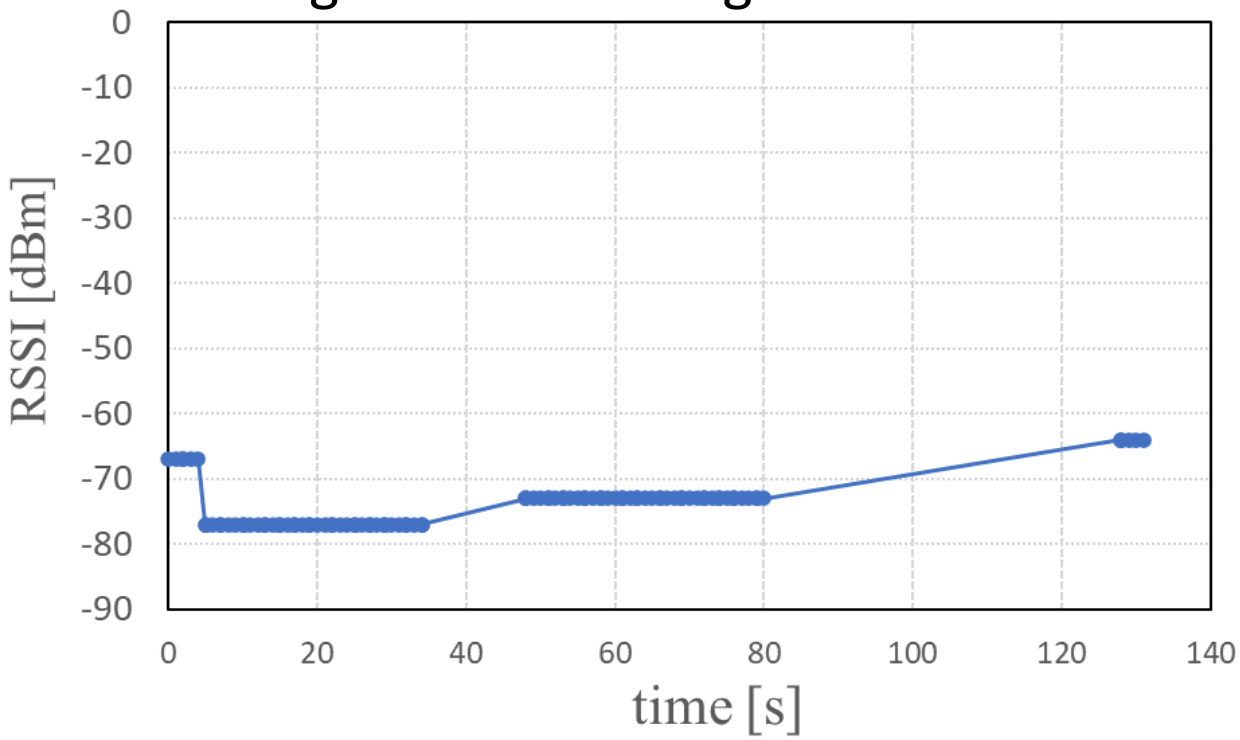
- Location estimation using only RSSI is difficult
  - Multi-Pass fading





- AI/ML is used for location estimation in this challenge

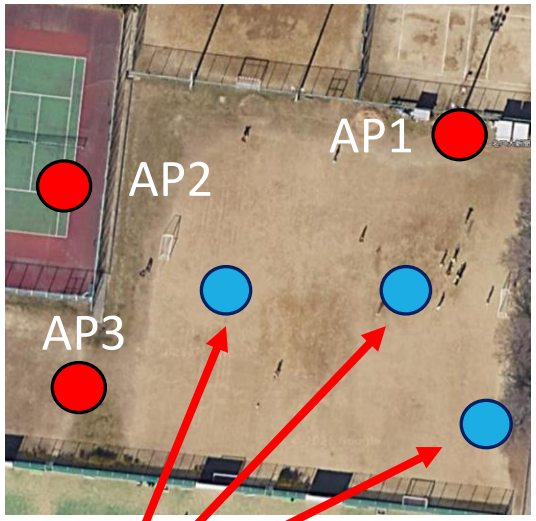
Change in received signal revel



Motivation

- However, it is still difficult to estimate a location using only RSSI
- Estimation error is not small even if AI/ML is used.
  - In our preliminary experiment, the maximum error is over 15 m.
- Moreover, for some context-aware applications, the determination of location is not needed, just the determination of area is adequate.
- We consider the area estimation in addition to the location estimation.

Location estimation



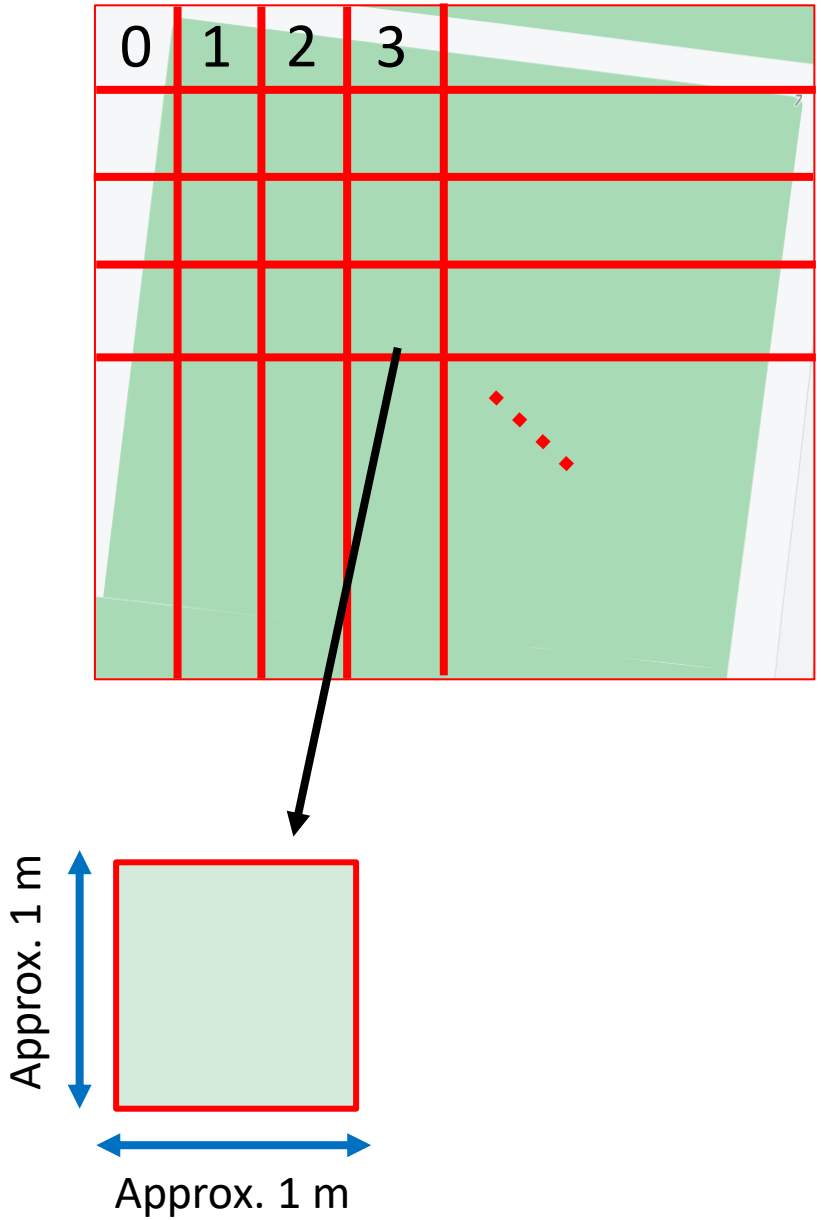
AP	RSSI
#1	Value 1
#2	Value 2
#3	Value 3

Area estimation

Proposed Method

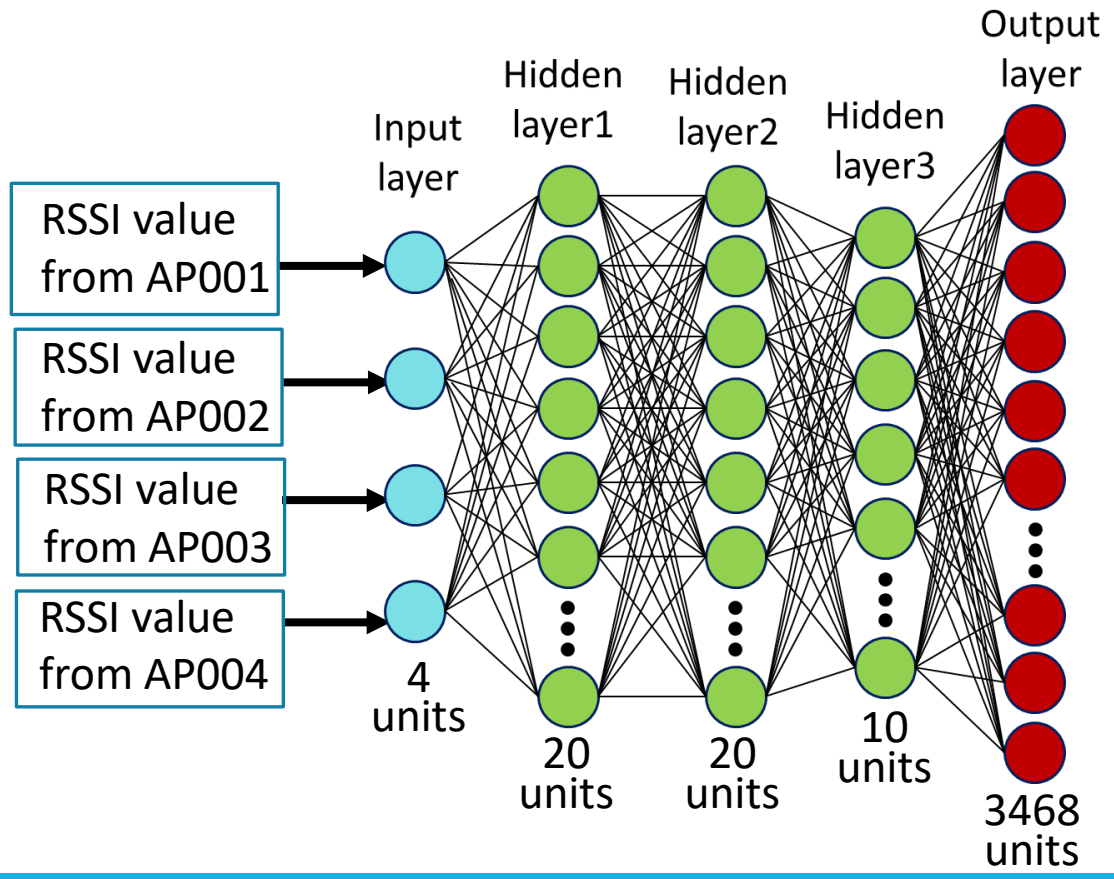
- Multi-Class Classification using DNN
  - Divide the map into smaller areas
  - Our DNN model predicts which area contains the measurement point

Area No.	Probability
13	40%
35	30%
34	11%
	⋮



Our Model

- Input: RSSI values from 4 points (AP001-AP004)
- Output: Probability with that the measurement point is included in each area



Output Area No.35

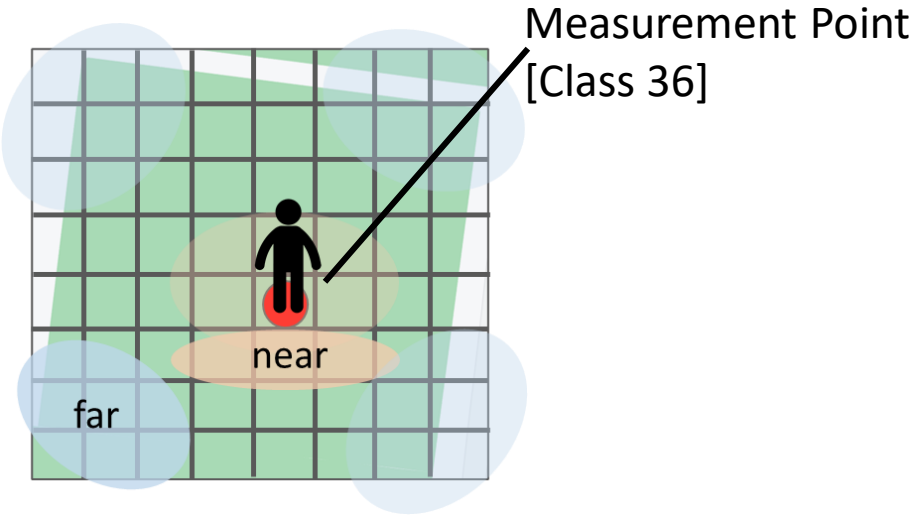
We select the area with highest probability as a estimation result

- Activation function

Layer	Activation function
Hidden layer1	ReLU
Hidden layer2	ReLU
Hidden layer3	ReLU
Output layer	softmax

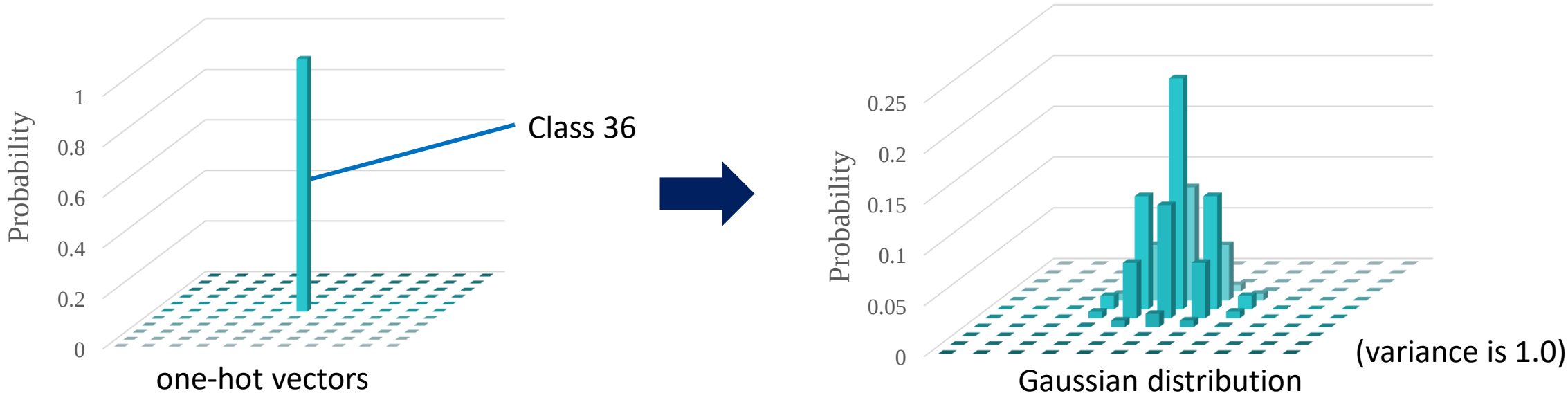
- Loss function: categorical cross entropy
  - Categorical cross entropy is widely used as loss function for multi-class classification problems.

Label Distribution Learning



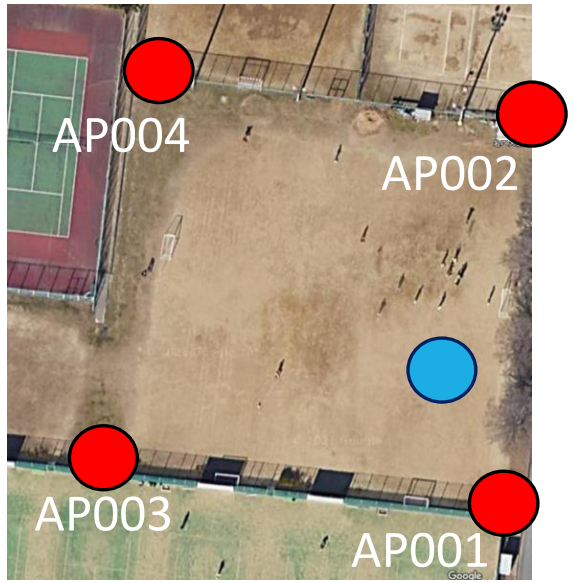
- There is an ordering among the classes to be classified.
  - We use categorical cross entropy as a loss function.
    - Whether the model predicts a near or far class, the loss is the same…

- Change the label (one-hot vectors) into a Gaussian distribution with the class label as the center(mean).
  - We use this distribution as a label.



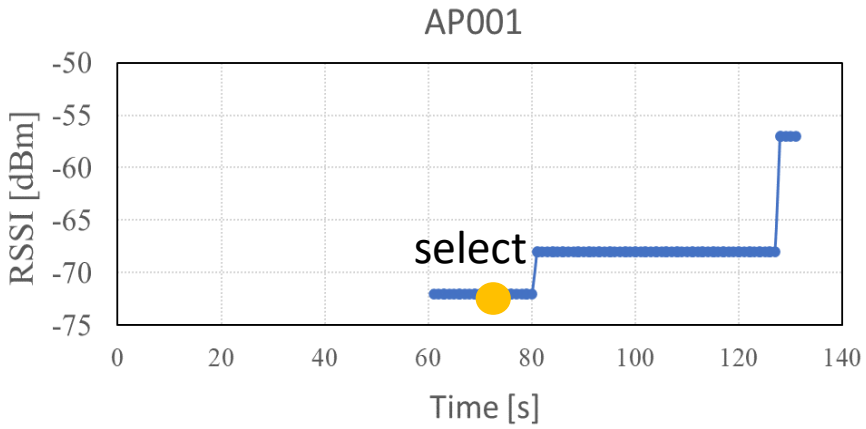
Generating Training Data

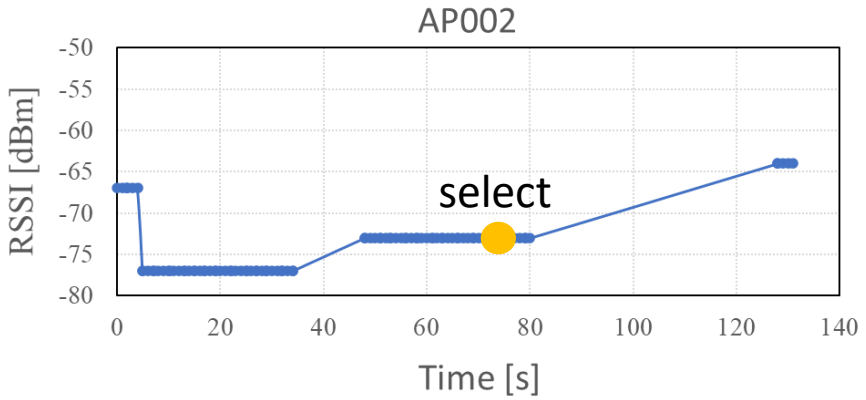
- We randomly select one value from the RSSI of each access point.
- Repeat random selection 500 times to generate 500 records of training data per measurement point

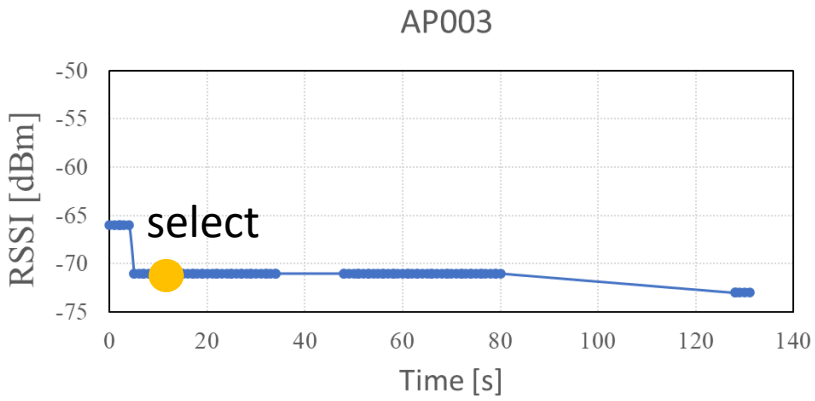


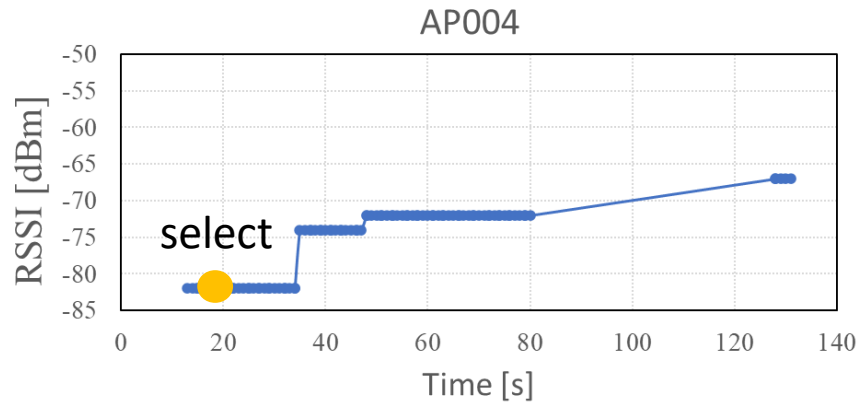
Access Point

Measurement Point

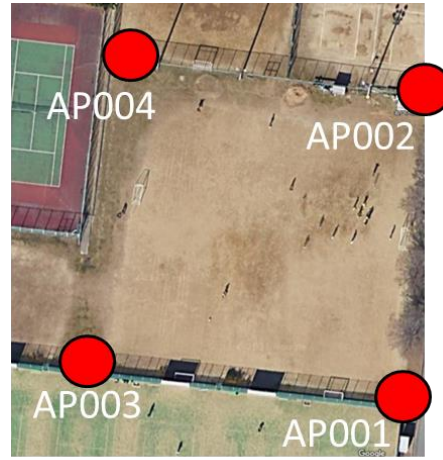




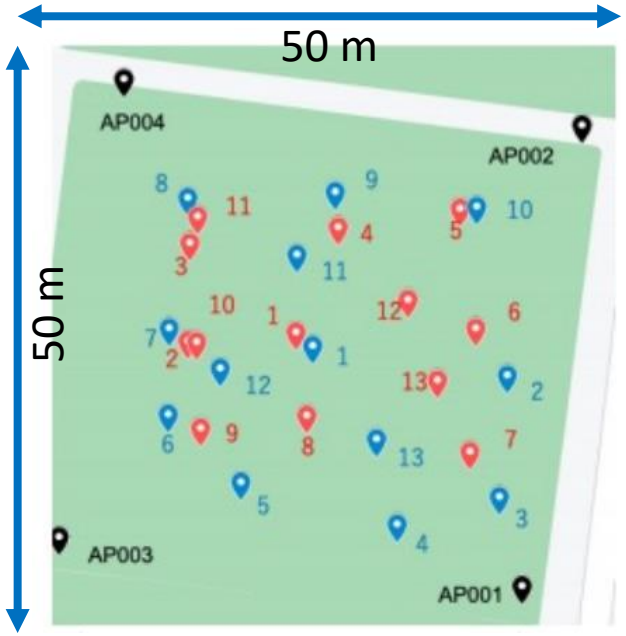




Localization results



- Data set 1



- Blue position : label of training data
- Red position : label of verification data

1.Average error: 11.1579971 m

2.Maximum error: 24.1497287 m

- Data set 2

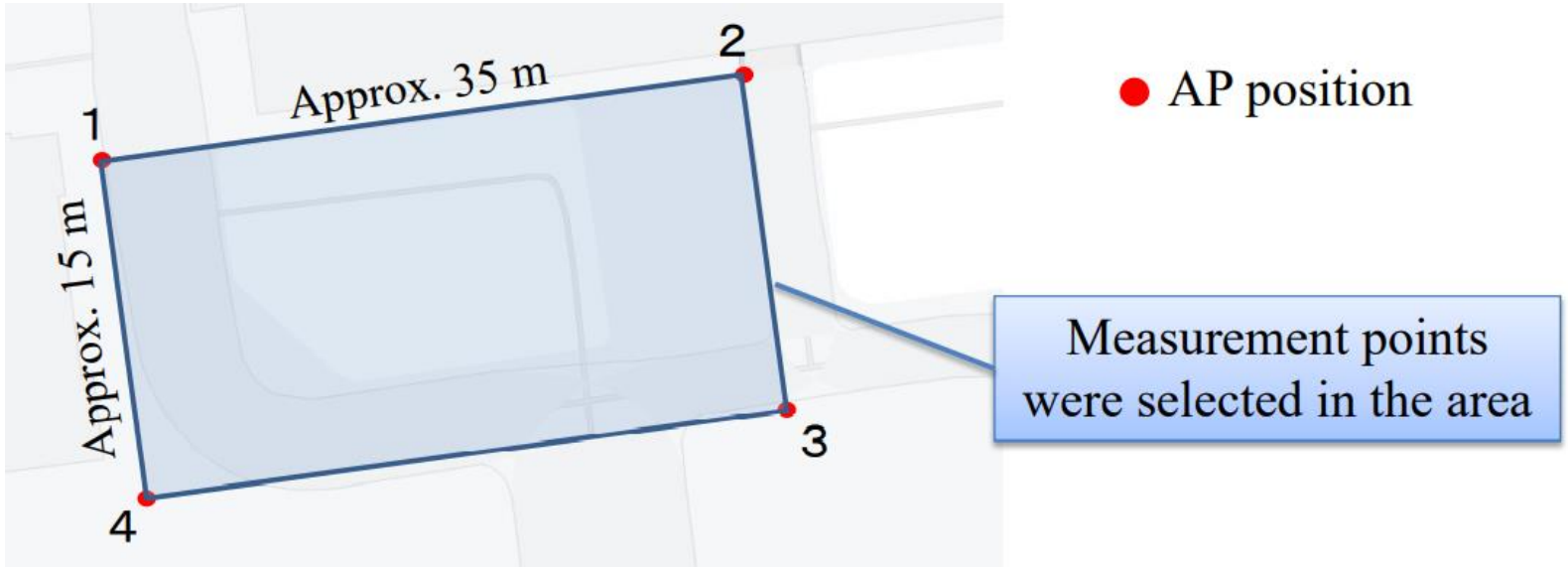
- Green position : label of training data
- Yellow position : label of verification data

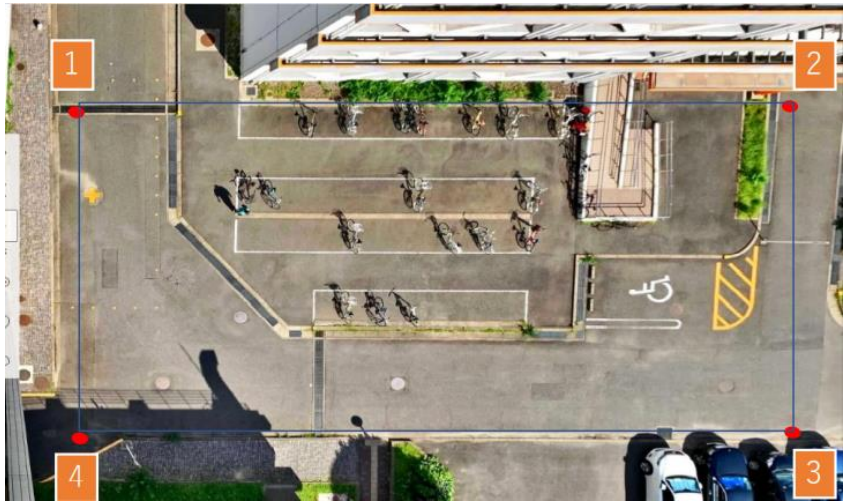
1.Average error: 11.8099257 m

2.Maximum error: 27.1292539 m

Localization results

- Data set for Grand Challenge Finale



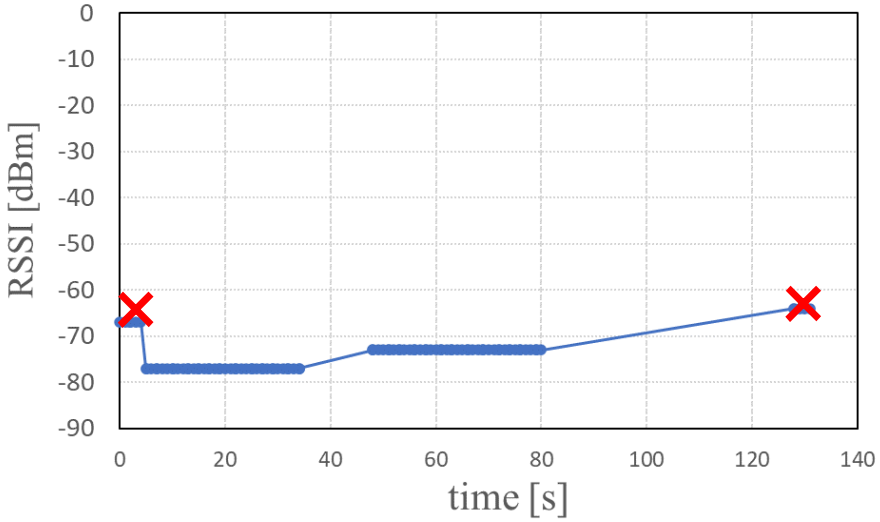


1.Average error: 13.1439172 m

2.Maximum error: 22.3437030 m

Discussion

- Localization results were not so good
- We pre-processed on the contents of data very little
  - Preprocessing may improve the accuracy.
    (e.g. delete data whose RSSI value is extremely different from others)



- Should we create a model that takes into account the time series of the data?
  - Our model did not take into account the time series.