Neural Network ODL (On-Device Learning) for Wireless Sensor Nodes

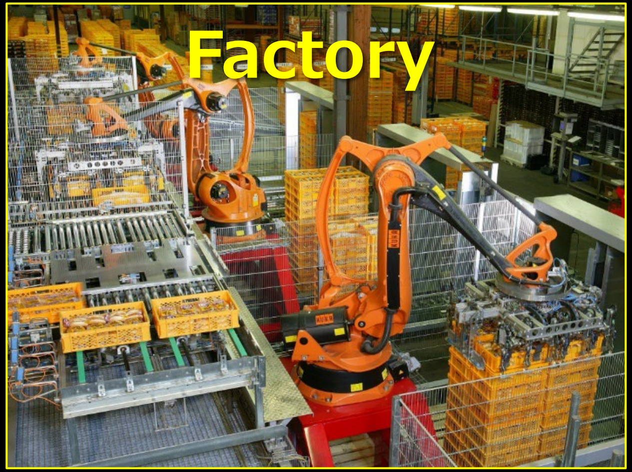
Hiroki Matsutani (Keio University, Japan)

@tinyML On Device Learning Forum (Aug 31, 2022)

IoT: Applications

- **ML application in real fields (e.g., anomaly detection)**

  Factory, monitoring, robot, safety, security, surveillance, …


Factory


Building

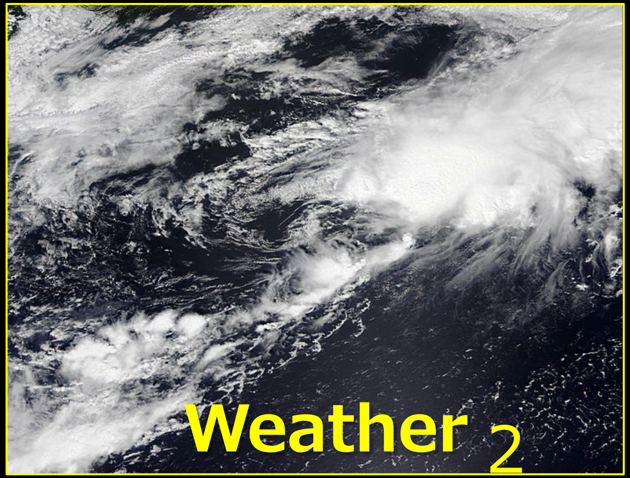

Robot (UAV)


Wheelchair


Electrical Safety

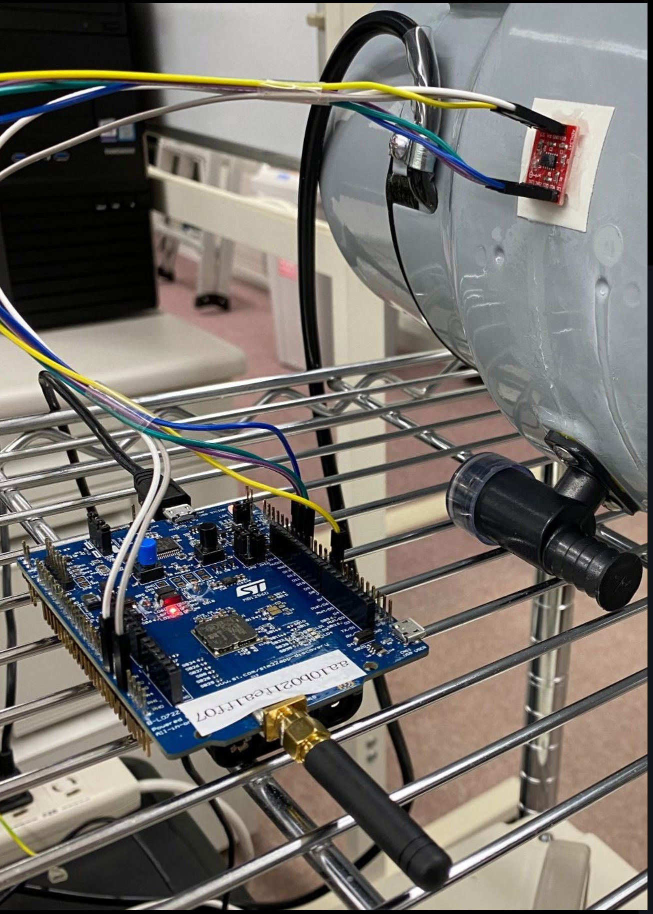

Surveillance

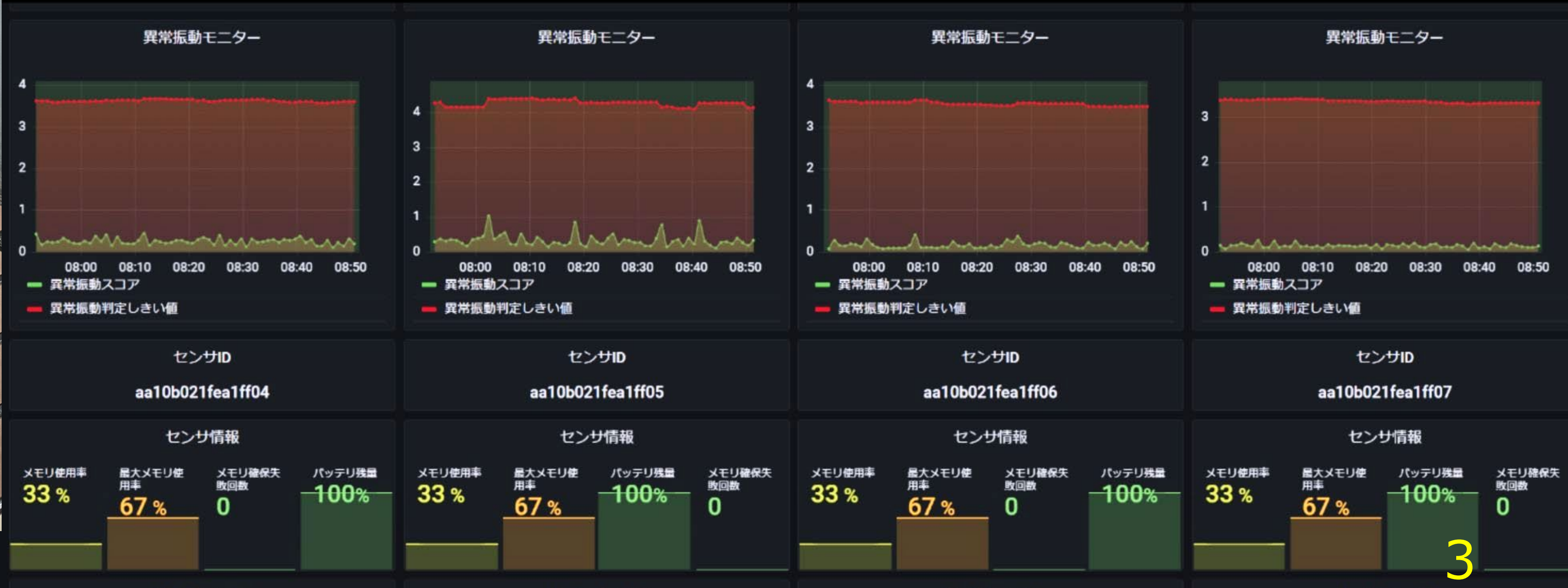

Body Movement


Weather

Edge AI: Equipment monitoring

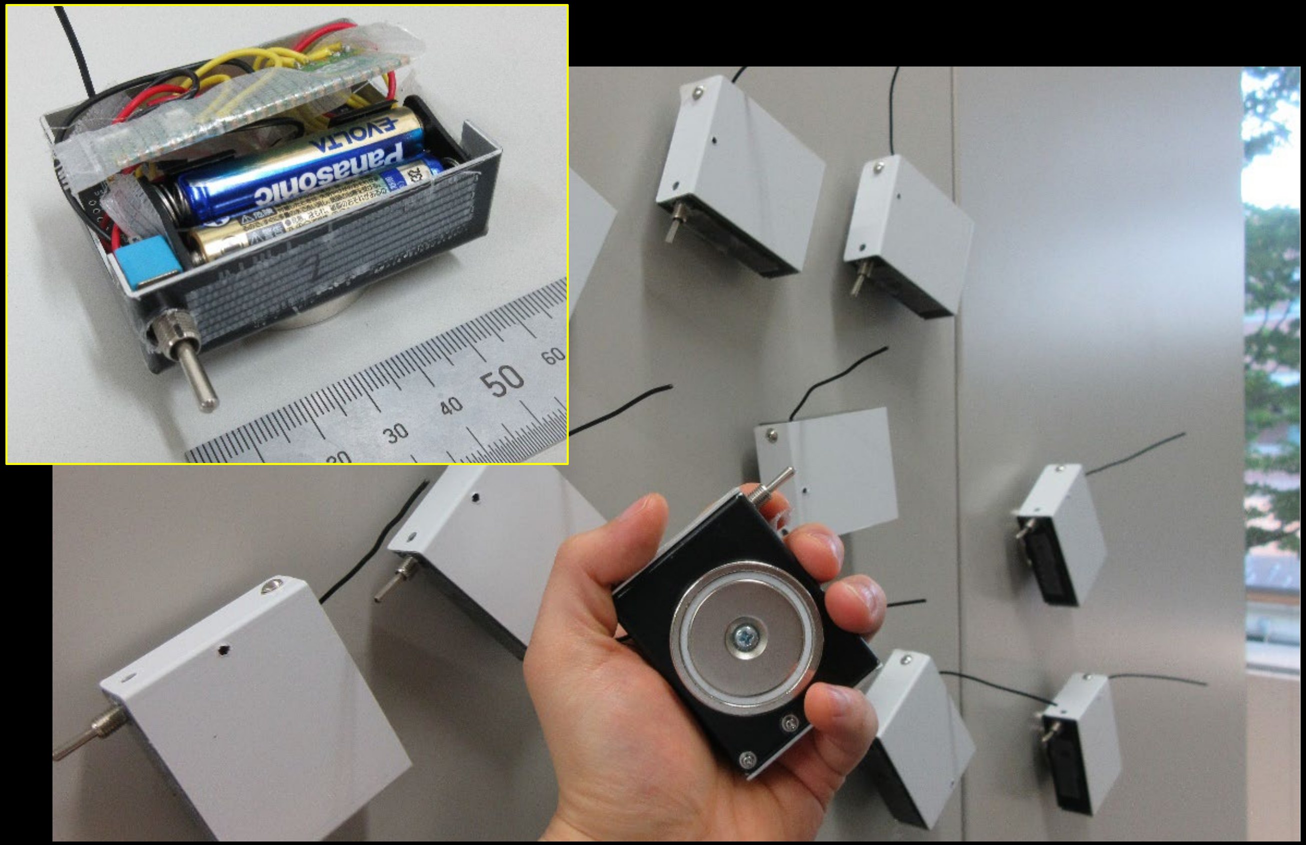
- **Monitoring of air-conditioning systems (e.g., fans)**
  Using wireless sensor nodes that can train and predict





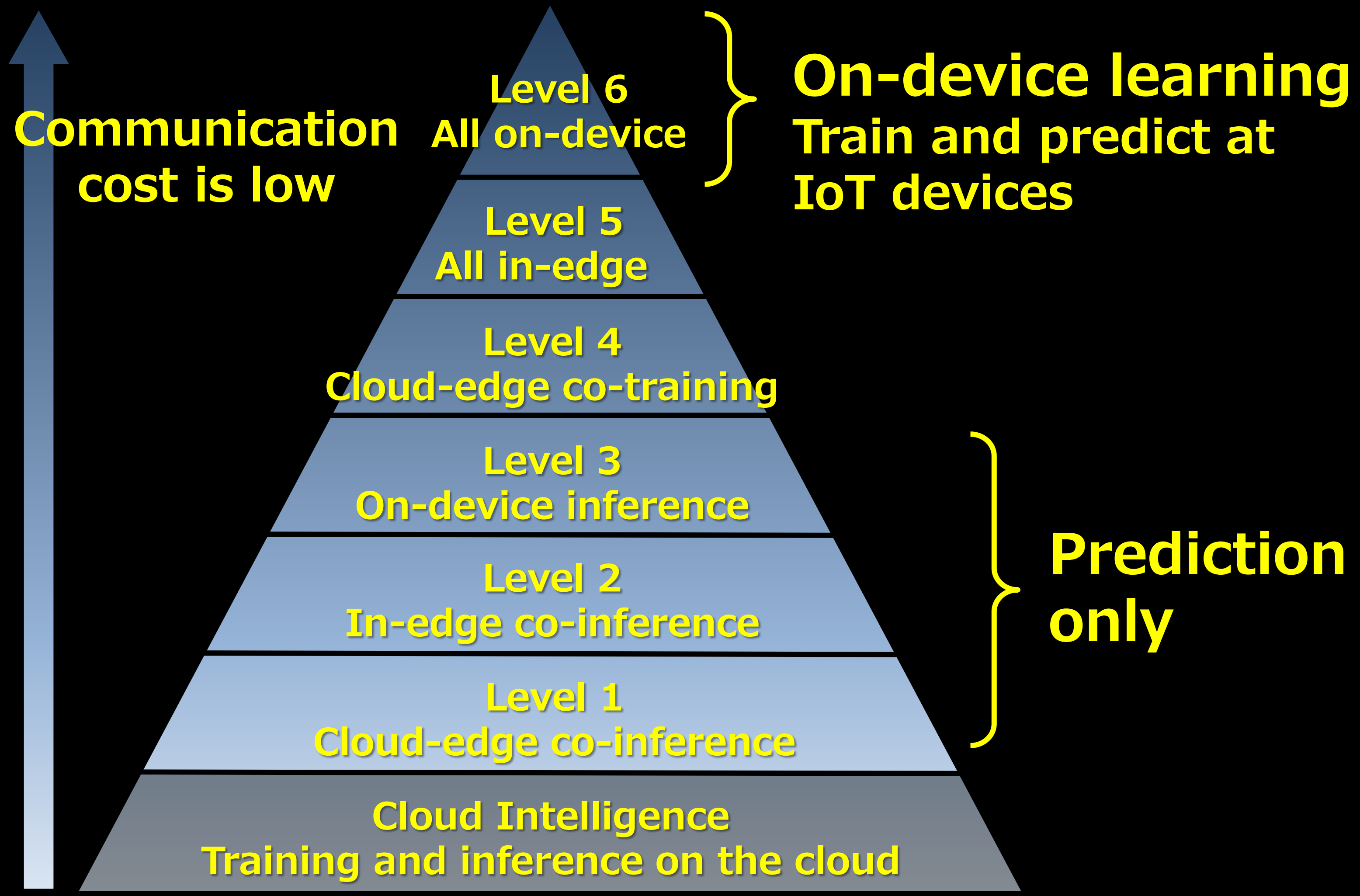
Edge AI: Equipment monitoring

- Wireless sensor nodes that can train and predict [1]
  - Raspberry Pi Pico, sensors, magnet, battery, LoRa module
  - On-device learning of neural networks

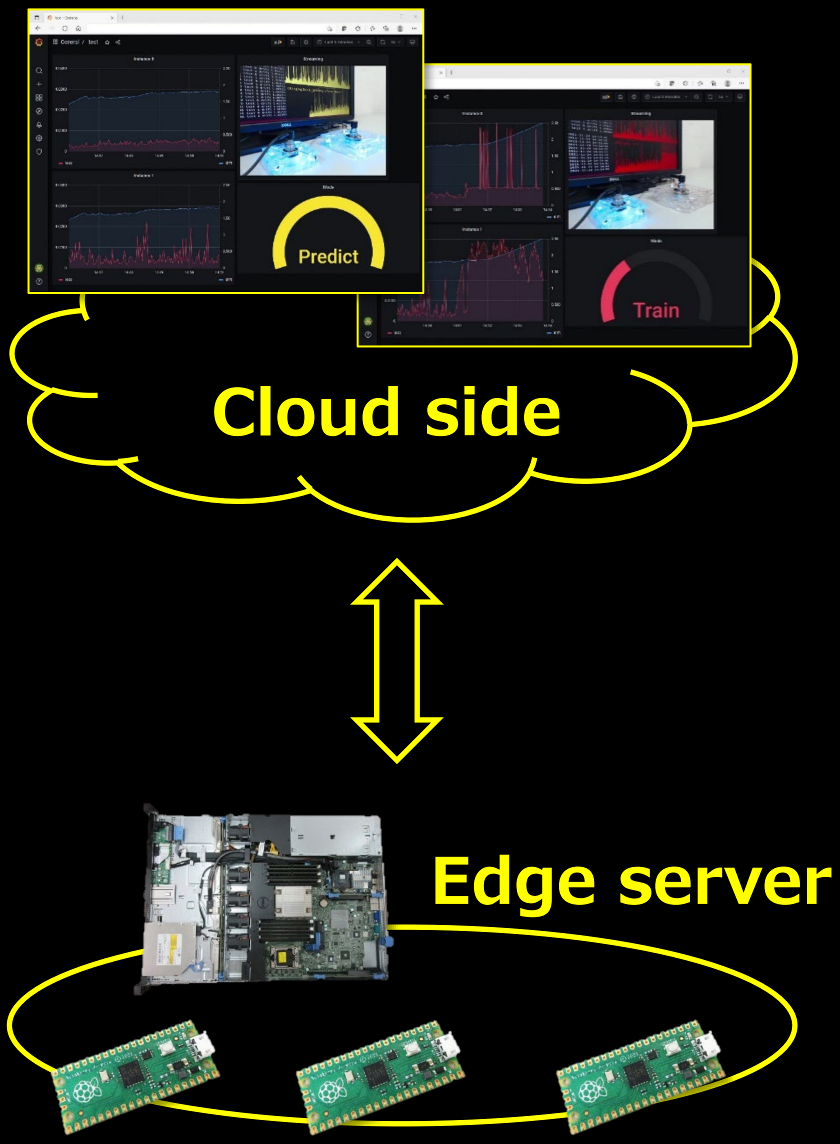


[1] Hiroki Matsutani et al., "On-Device Learning: A Neural Network Based Field-Trainable Edge AI", arXiv:2203.01077 (2022).

Edge AI: Classification



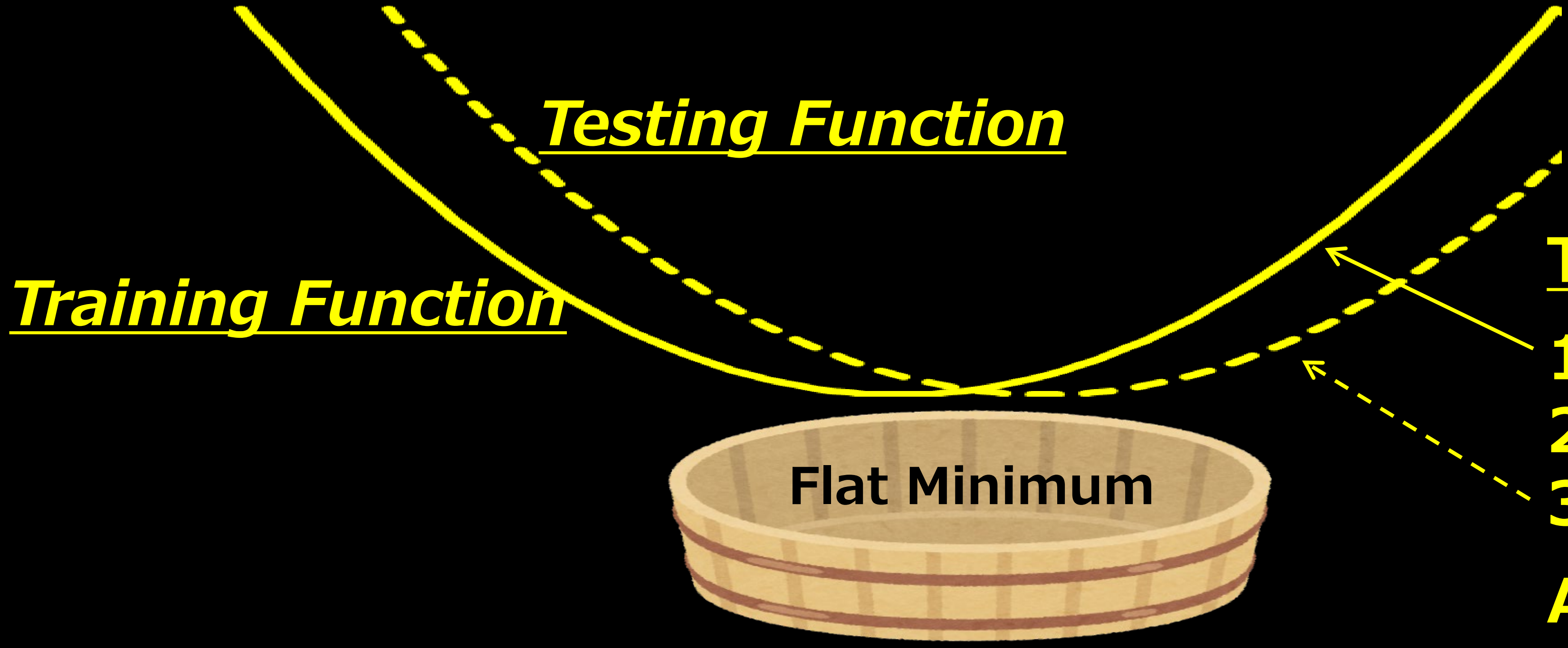
Six-level rating for edge intelligence [1]



[1] Z. Zhou et al., "Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing", Proceedings of the IEEE (2019).

On-device learning: Motivation

- Challenges of edge AI: Addressing the gap between training data and deployed environment



Typical edge AI use case

1. Collect train data
2. Train at server
3. Predict at edge

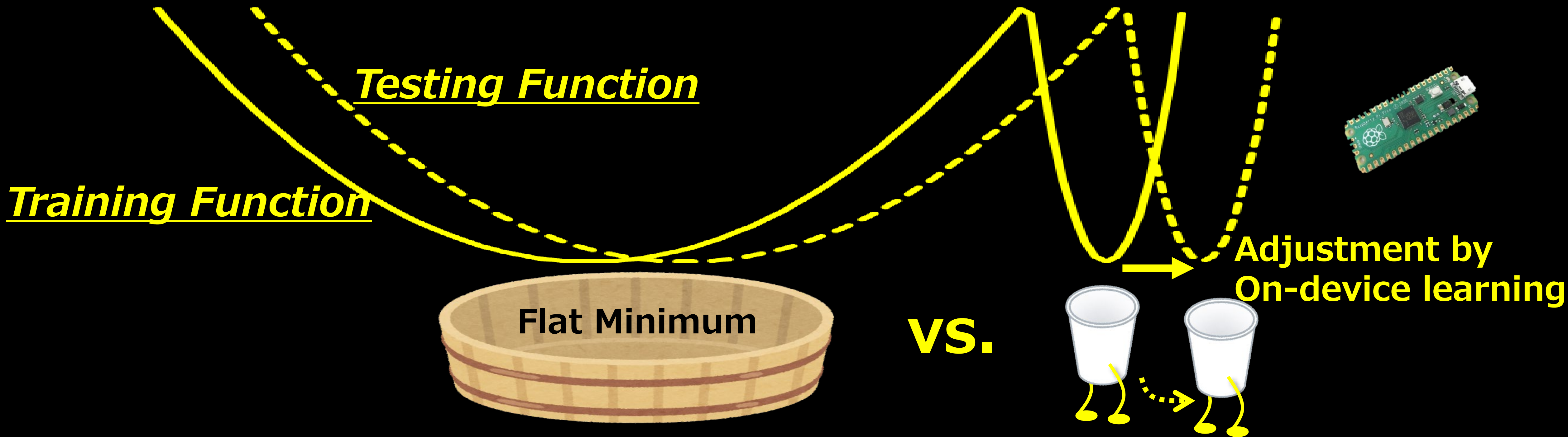
At different environments

Typical solution

✓ Generalization capability to absorb the gap

On-device learning: Motivation

- Challenges of edge AI: Addressing the gap between training data and deployed environment at low-cost



Typical solution
- ✓ Generalization capability to absorb the gap

Our approach [1]
- ✓ Small neural networks
- ✓ Train at deployed environment

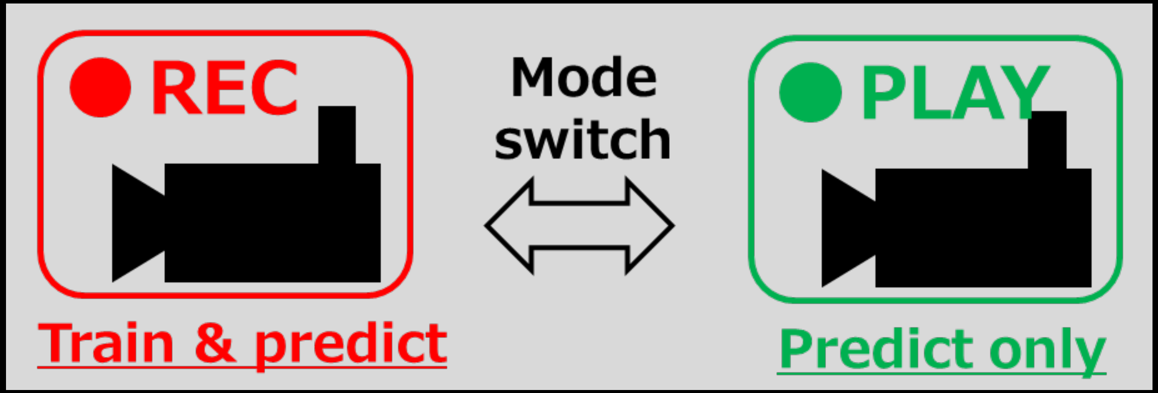
[1] Mineto Tsukada et al., "A Neural Network-Based On-device Learning Anomaly Detector for Edge Devices", IEEE Trans. on Computers (2020).

On-device learning: Two modes

1. Train mode

2. Predict-only mode

Question: ***How and when is the mode changed?***



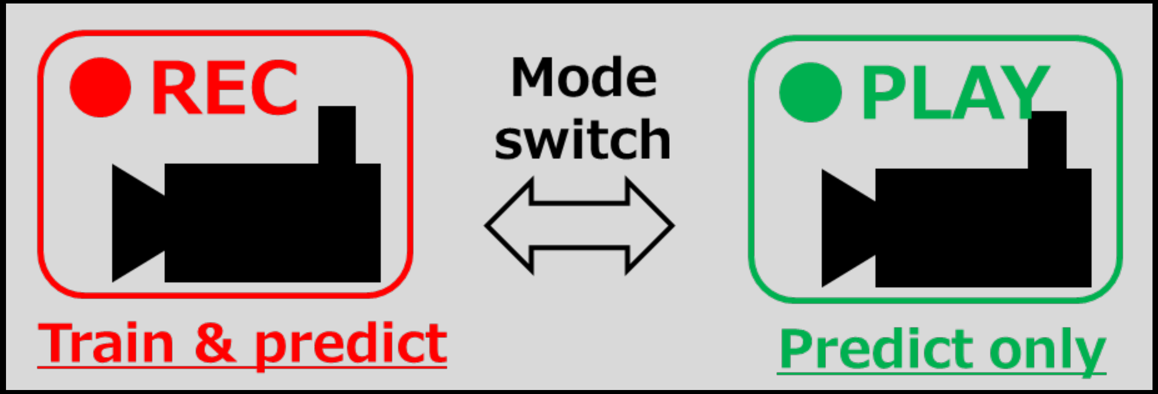
[1] Hiroki Matsutani et al., "On-Device Learning: A Neural Network Based Field-Trainable Edge AI", arXiv:2203.01077 (2022).

On-device learning: Two approaches

1. Field-tunable AI

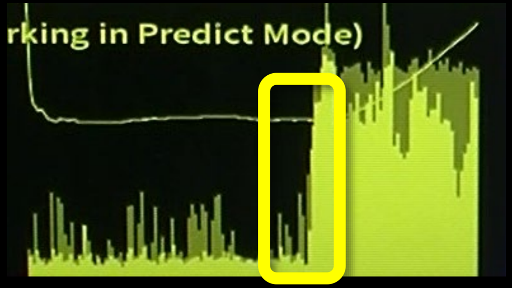
Field-engineers can train edge AI whenever they want

Triggered by train button

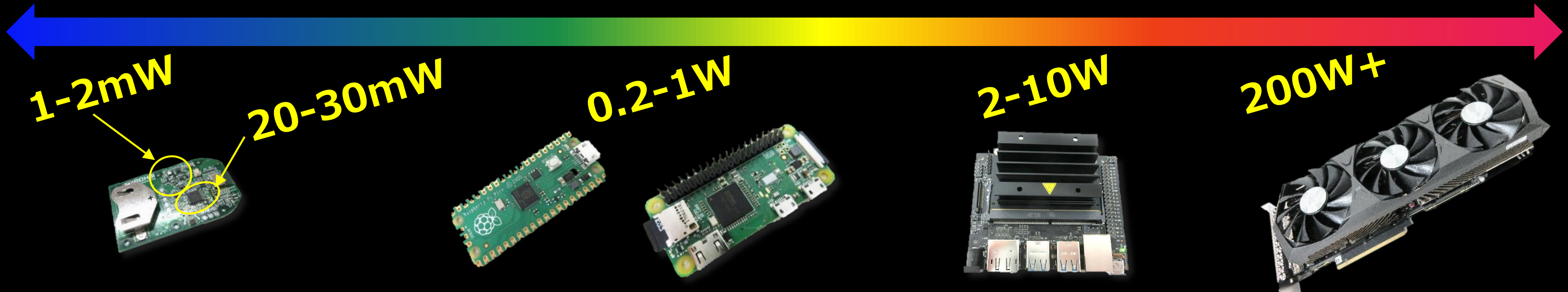


2. Field-adaptive AI

Automatically trained when concept drift is detected

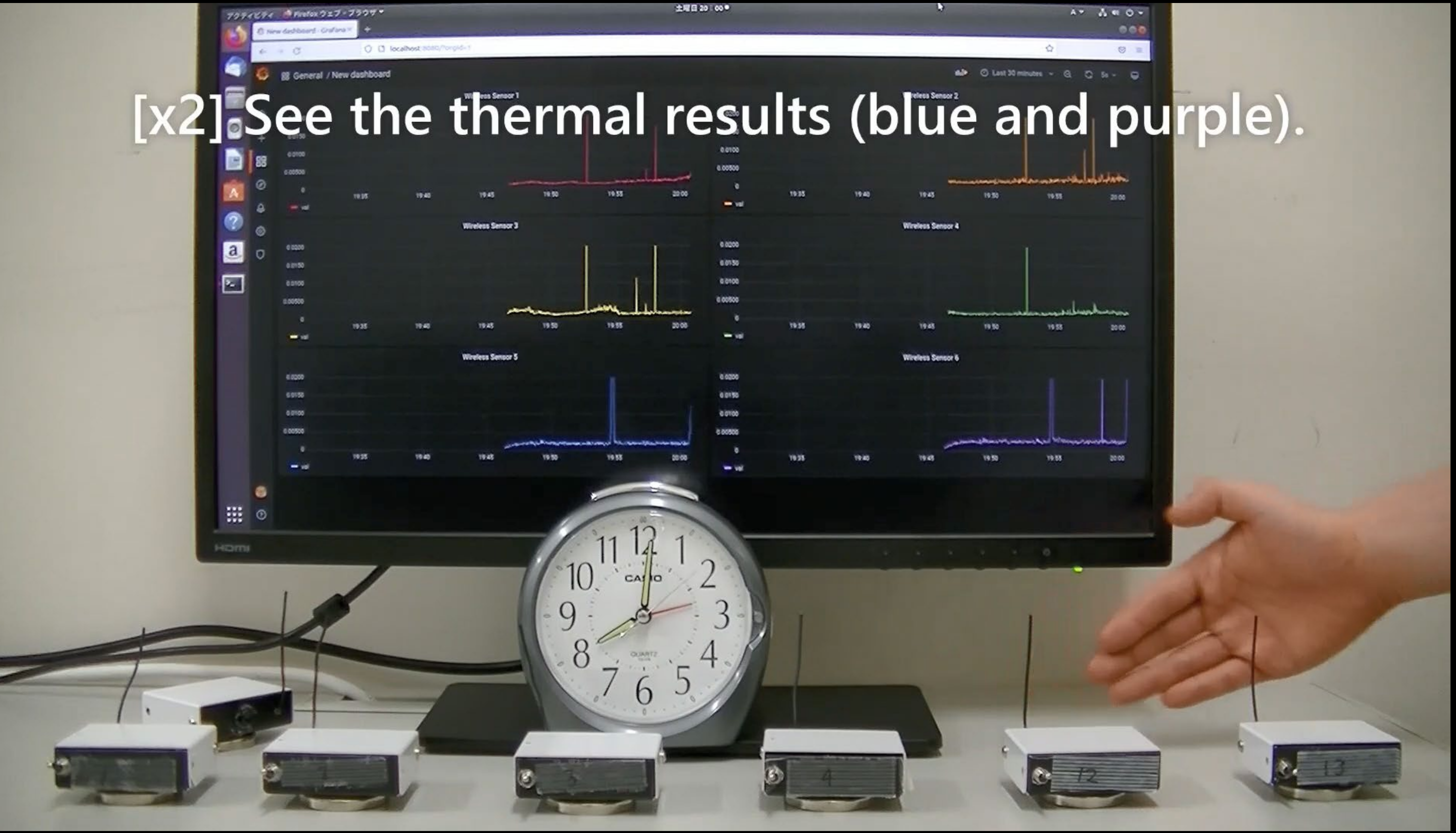


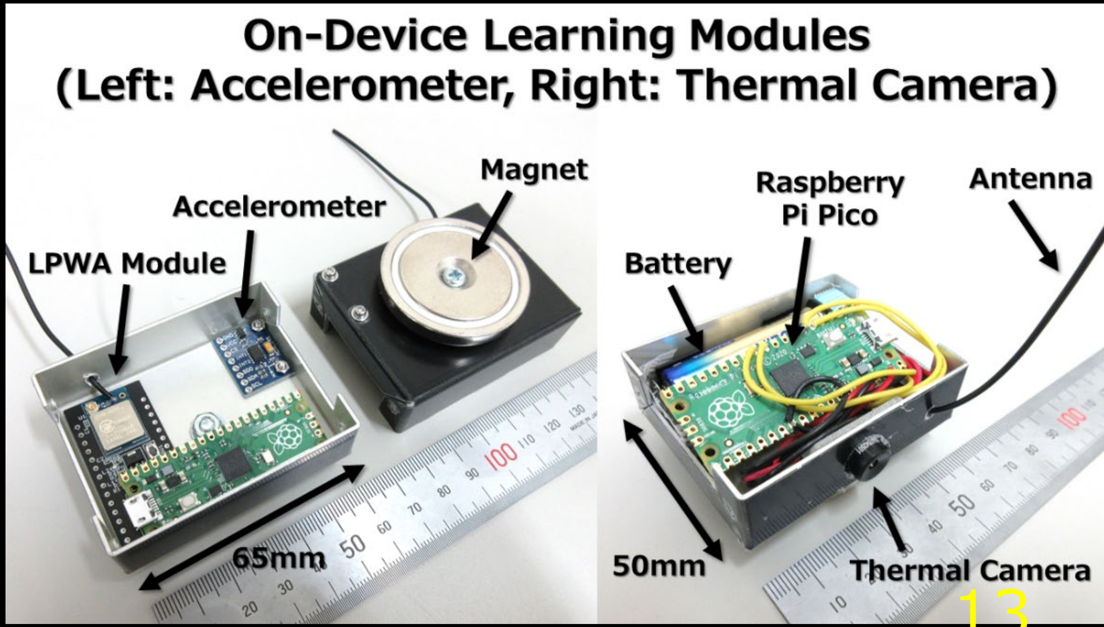
Triggered by concept drift



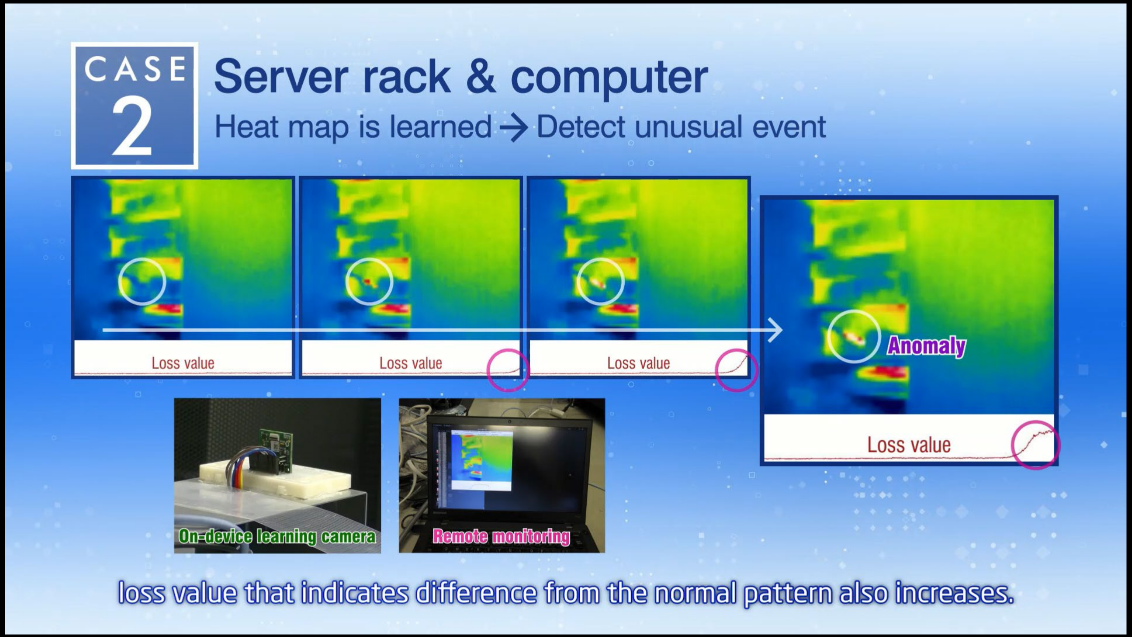
[1] Hiroki Matsutani et al., "On-Device Learning: A Neural Network Based Field-Trainable Edge AI", arXiv:2203.01077 (2022).

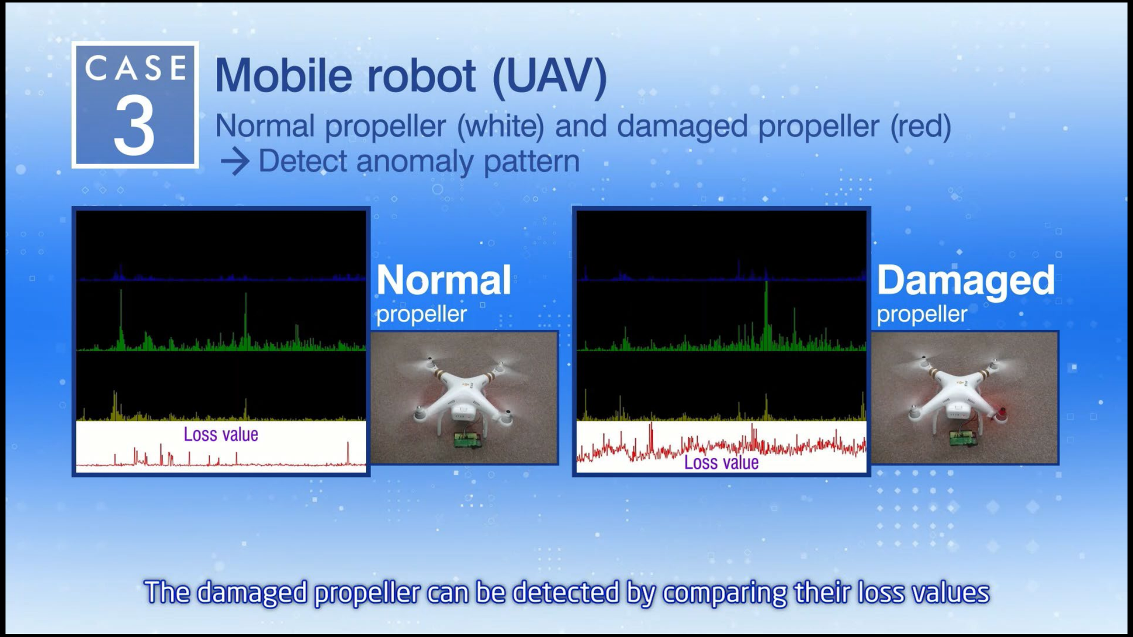
On-device learning: Demonstration





On-device learning: Other demos

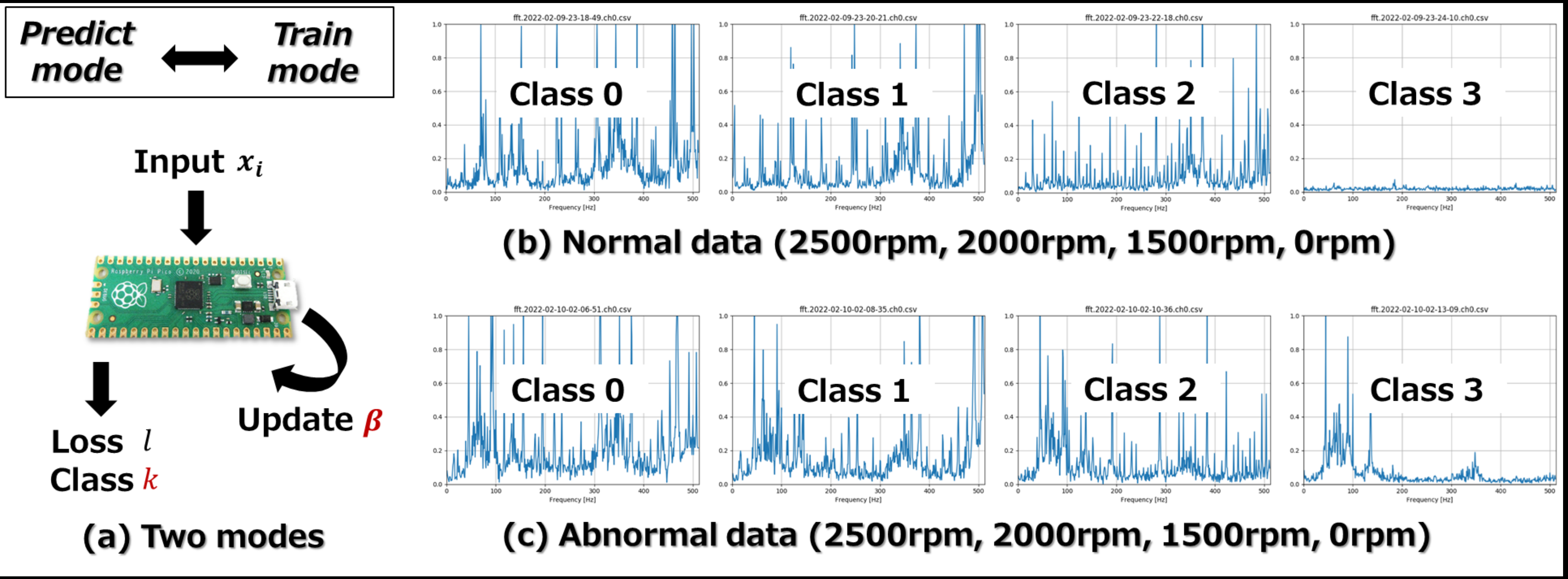




On-device learning: Algorithm

- *K*-class anomaly detection with *K* autoencoder neural networks (i.e., instances)

Each autoencoder is trained to be specialized to each class

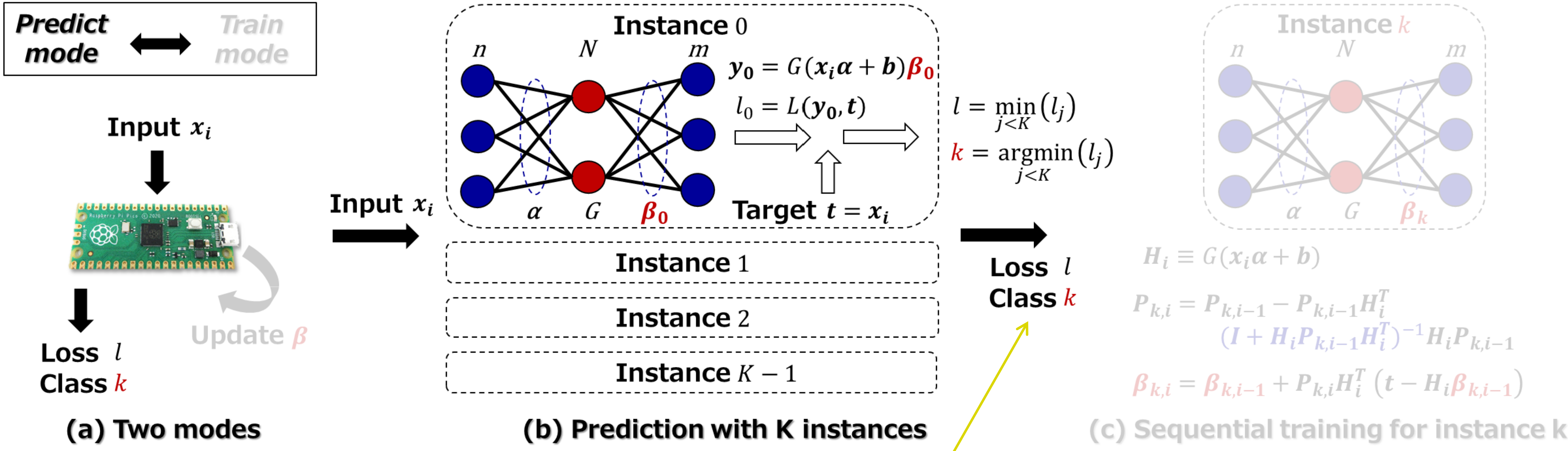


Example: 4-class anomaly detection on cooling fan dataset (mentioned later)

On-device learning: Prediction

- Prediction is done by *K* instances, each of which is specialized to each class

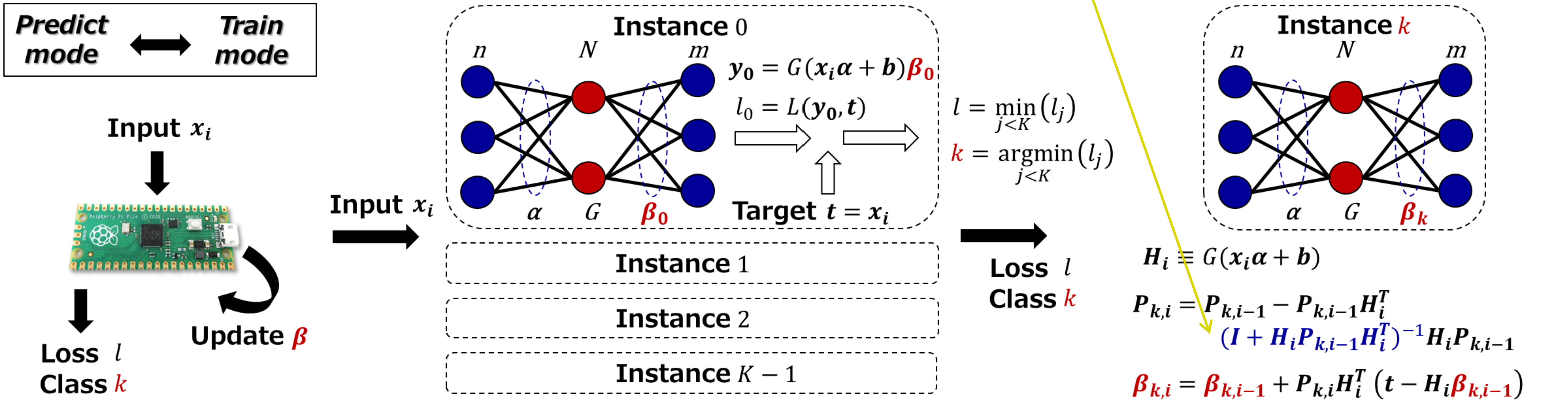
Input: *n*-dimensional data, Output: Loss *l* and class *k*



Instance with the smallest loss value is "the closest" instance or class

On-device learning: Sequential training

- "The closest instance" is updated with the input data
  - OS-ELM [1] is used as sequential training algorithm
  - Batch size is fixed to 1 to eliminate pseudo inverse operation



(a) Two modes (b) Prediction with K instances (c) Sequential training for instance k

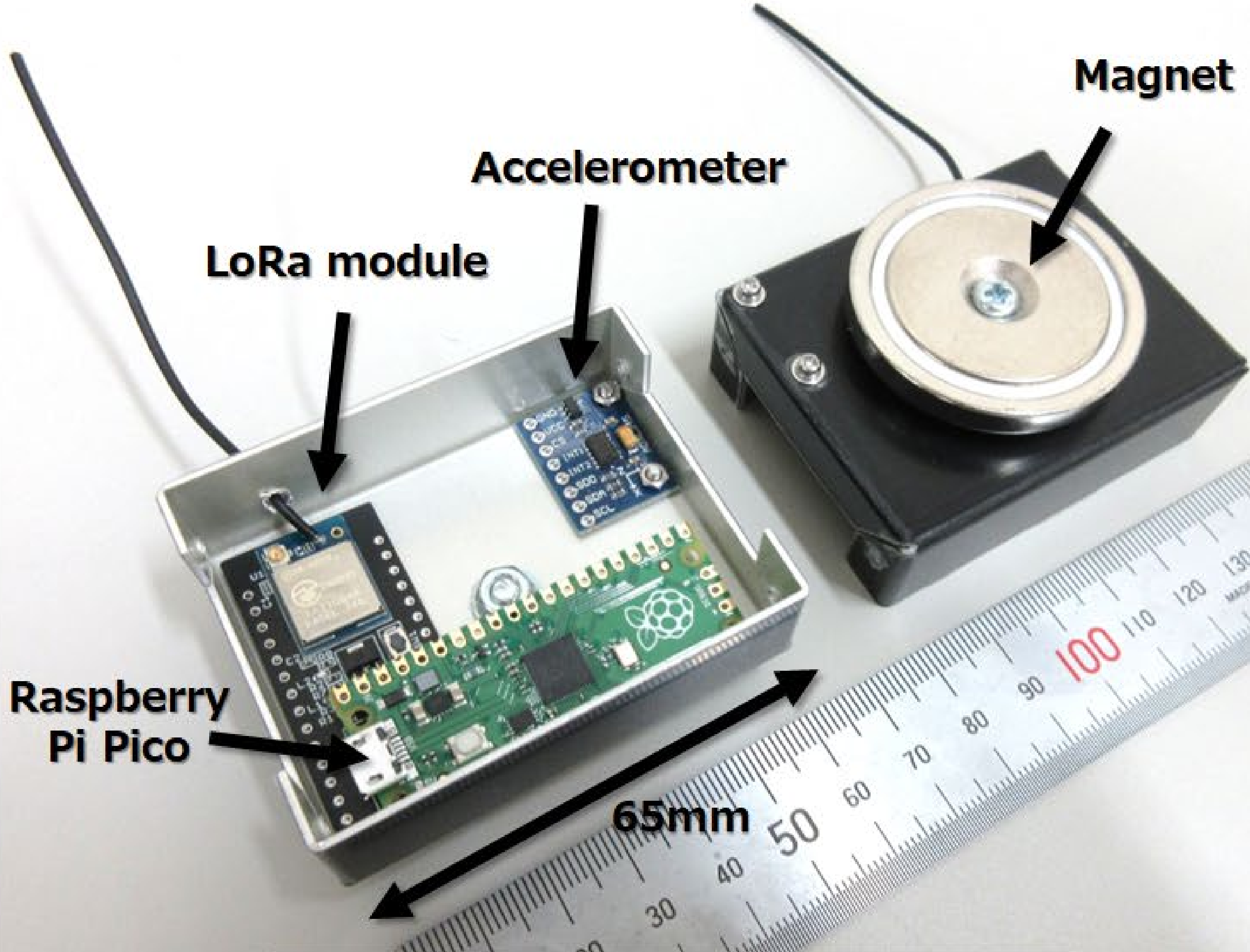
[1] N. Y. Liang, G. B. Huang, P. Saratchandran, N. Sundararajan, "A Fast and Accurate Online Sequential Learning Algorithm for Feedforward Networks", IEEE Trans. on Neural Networks, vol. 17, no. 6, pp. 1411-1423, Nov. 2006.

Evaluations: Execution time

- Raspberry Pi Pico @133MHz + LoRa module

(a) Execution time of on-device learning (total)

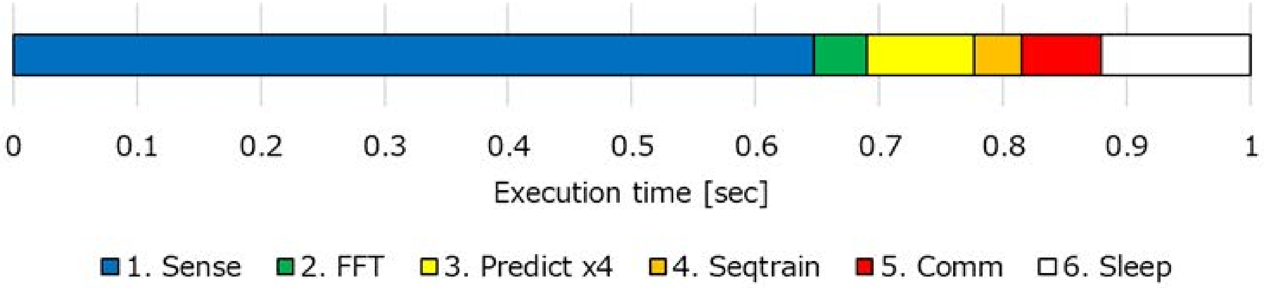
(b) Execution time of on-device learning (predict & train)



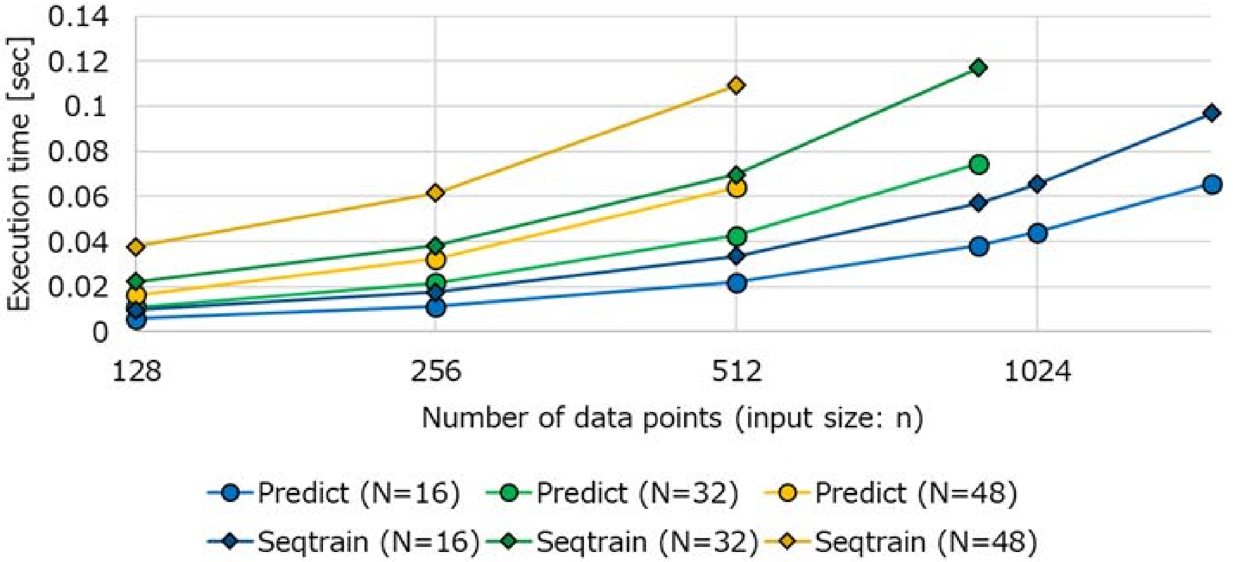
(c) Implementation of sensor node

Evaluations: Execution time

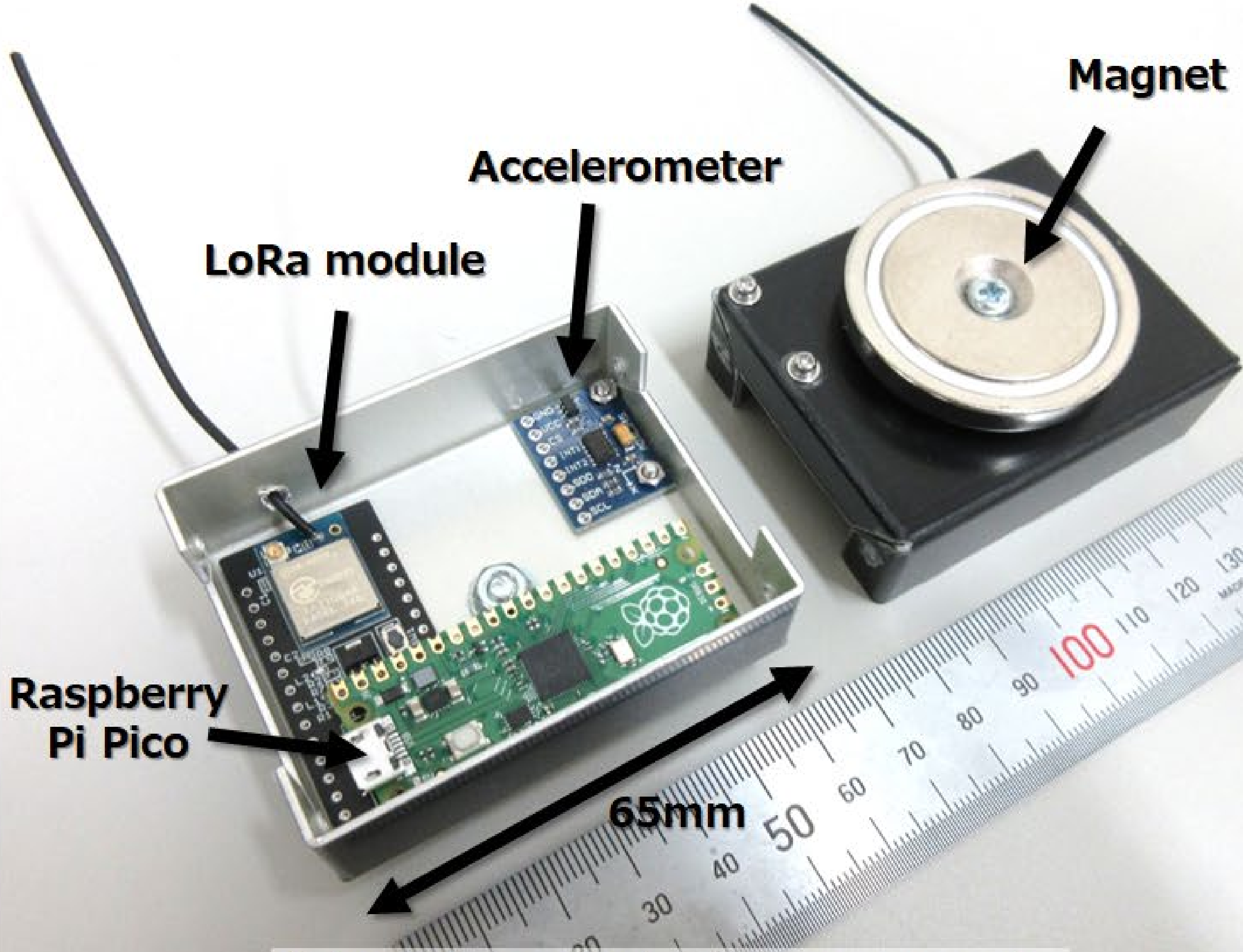
- Four 3-layer neural networks (256-32-256 nodes)



(a) Execution time of on-device learning (total)



(b) Execution time of on-device learning (predict & train)

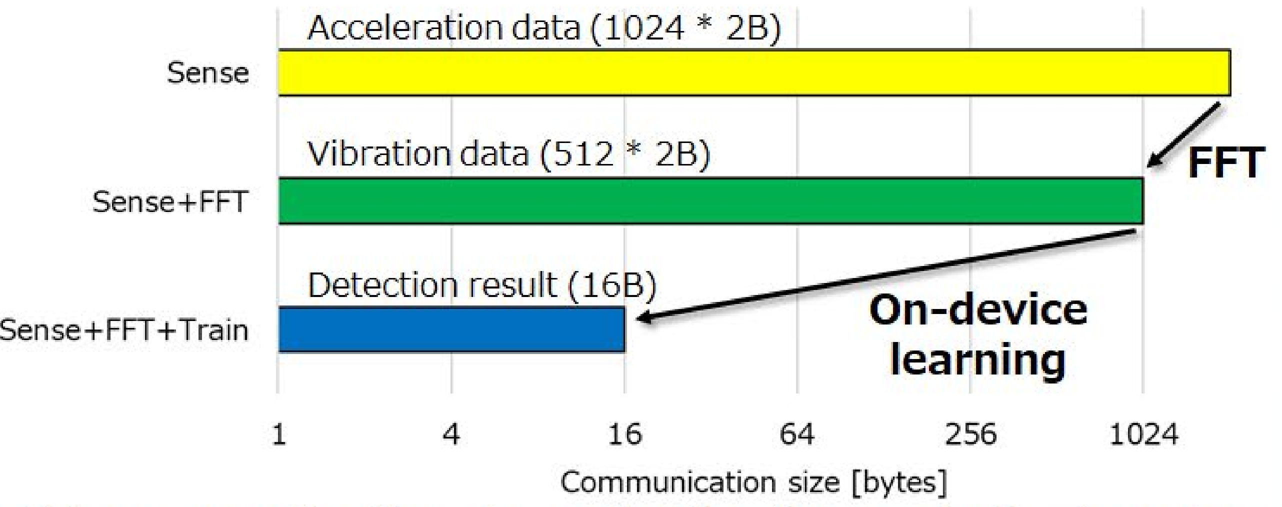


(c) Implementation of sensor node

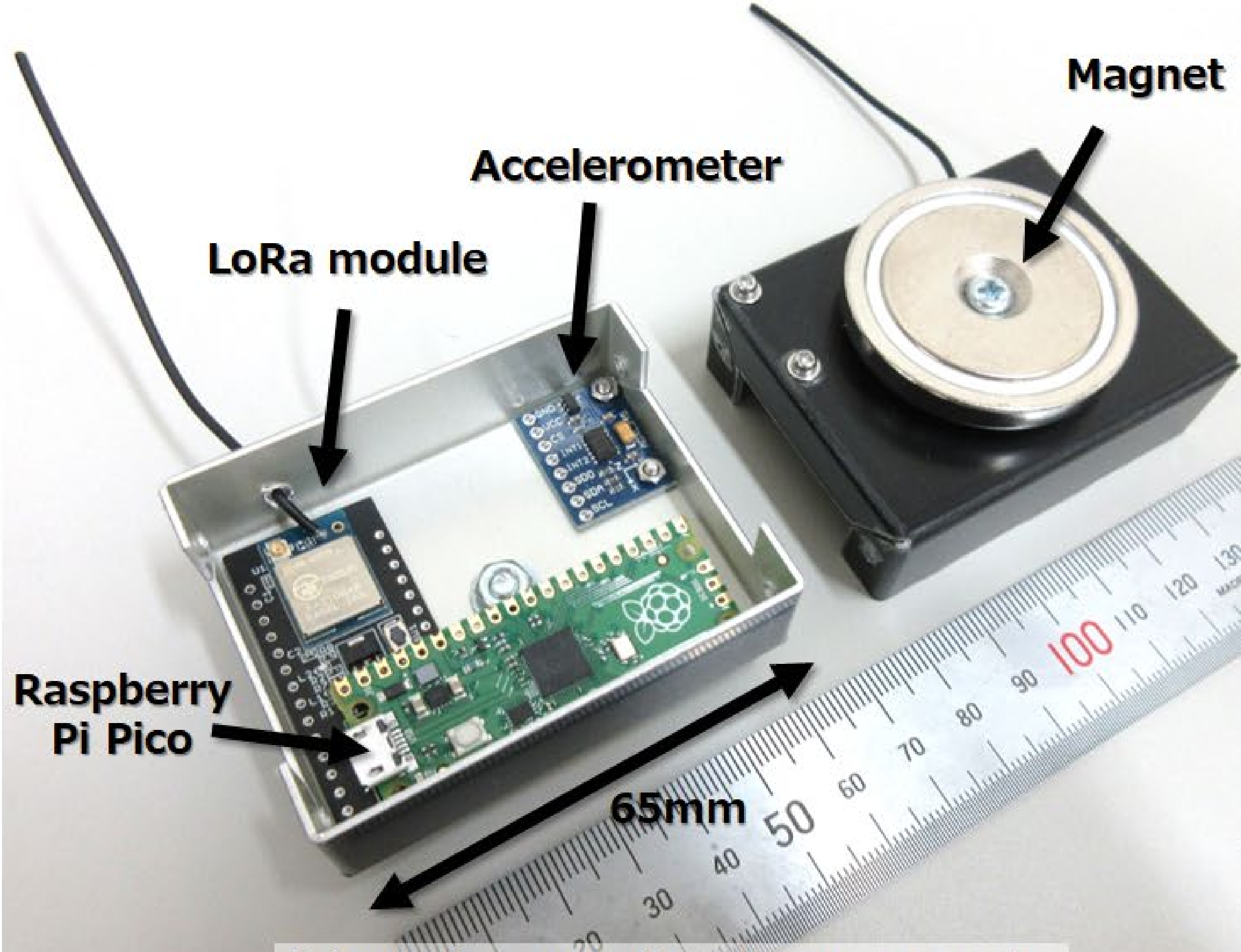
Evaluations: Communication size

- Communication size reduction not sending train data

(a) Execution time of on-device learning (total)



(b) Communication size reduction by on-device learning



(c) Implementation of sensor node

Evaluations: Cooling fan dataset

- Trained at normal but predict at noisy environment
  - Predict only: Cannot adapt to the noisy environment ☹
  - On-device learning: Can adapt to the noisy environment ☺

Big difference between training function and testing function

Normal environment → Noisy environment

Trained at normal environment

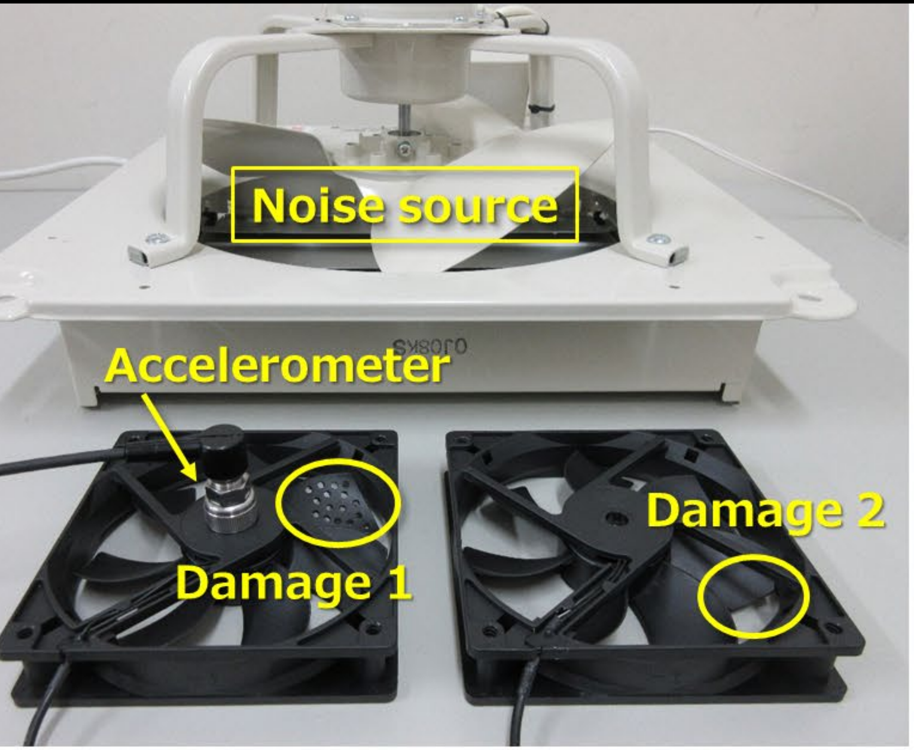
Deployed at noisy environment

Can anomaly detect?

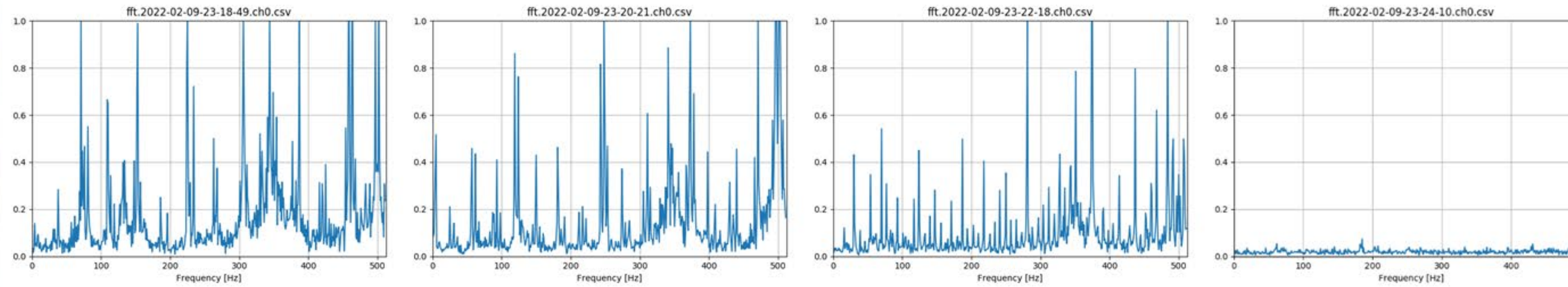
(Normal environment: office room, Noisy environment: near a ventilation fan)

Evaluations: Cooling fan dataset

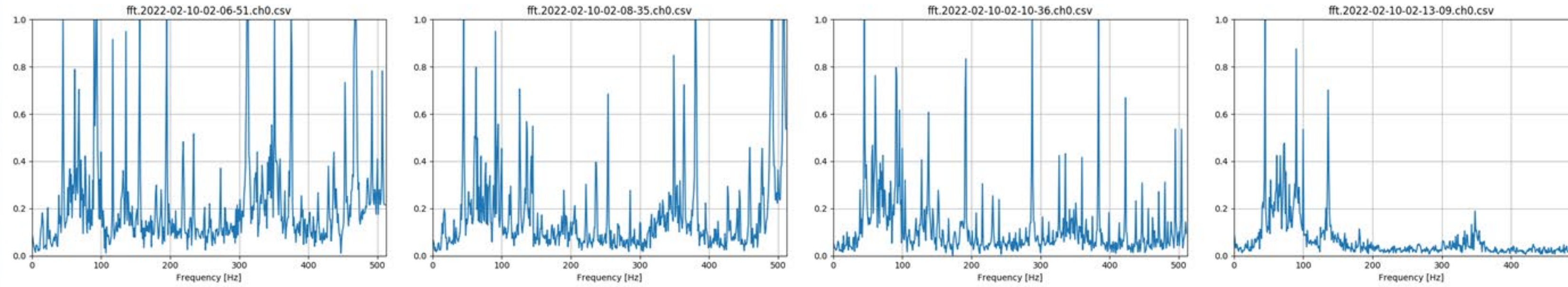
- Trained at normal but predict at noisy environment
  - Predict only: Cannot adapt to the noisy environment ☹
  - On-device learning: Can adapt to the noisy environment ☺



(a) Dataset measurement



(b) Normal environment (2500rpm, 2000rpm, 1500rpm, 0rpm)



(c) Noisy environment (2500rpm, 2000rpm, 1500rpm, 0rpm)

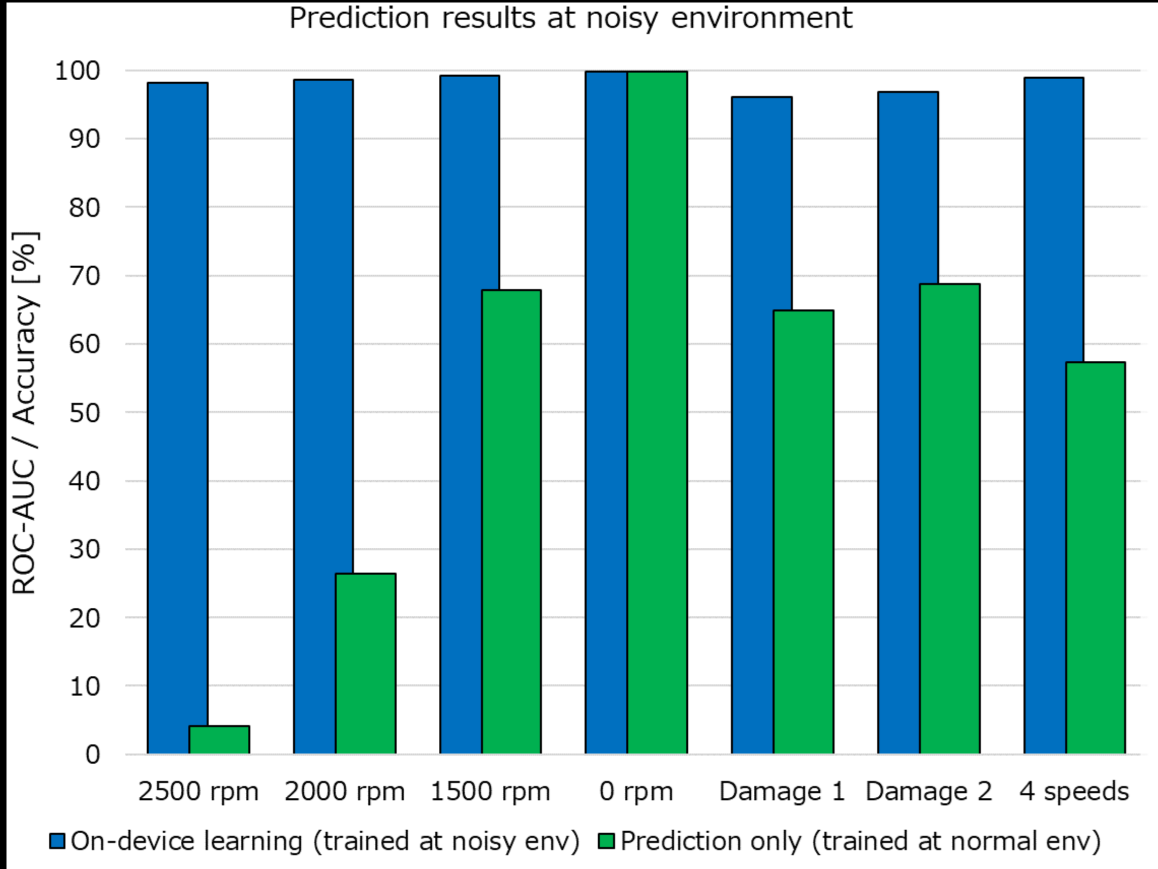
(Normal environment: office room, Noisy environment: near a ventilation fan)

Evaluations: ROC-AUC / Accuracy

- **Trained at normal but predict at noisy environment**

Predict only: Cannot adapt to the noisy environment ☹

On-device learning: Can adapt to the noisy environment ☺



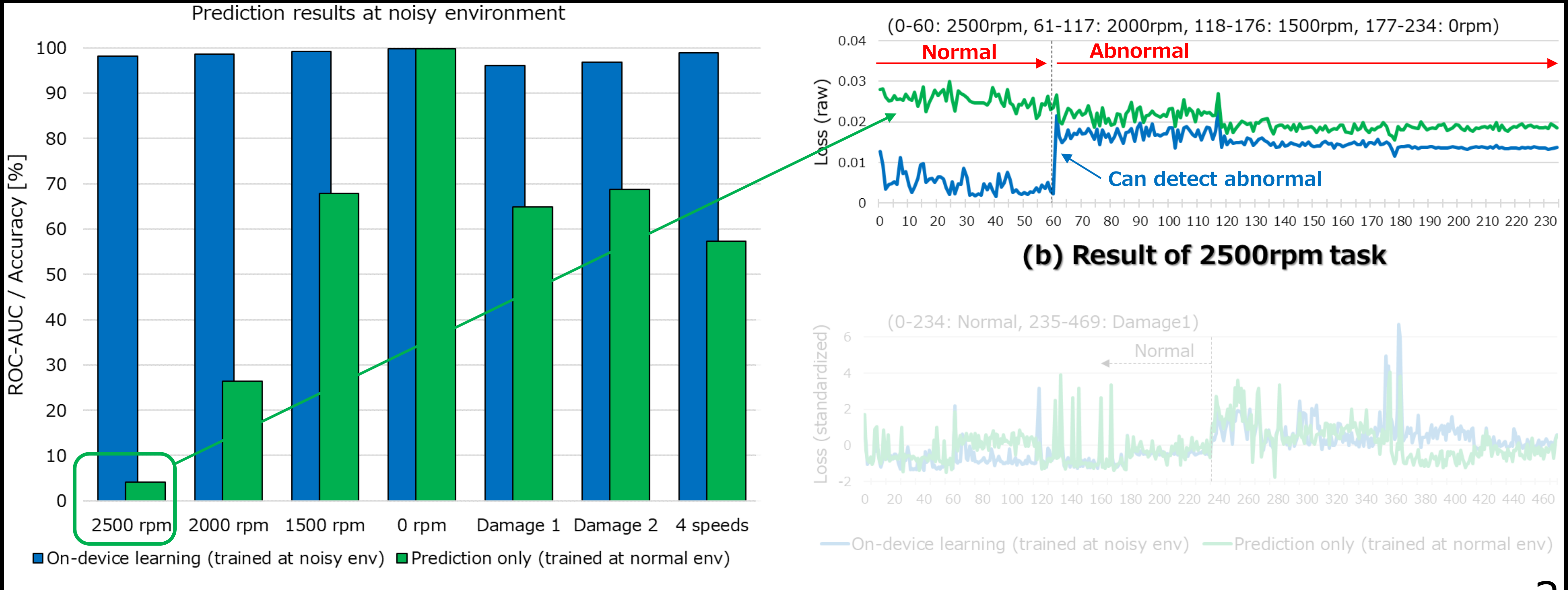
(a) Summary of seven tasks

(b) Result of 2500rpm task

(c) Result of Damage1 task

Evaluations: ROC-AUC / Accuracy

- Trained at normal but predict at noisy environment
  - Predict only: Cannot adapt to the noisy environment ☹
  - On-device learning: Can adapt to the noisy environment ☺



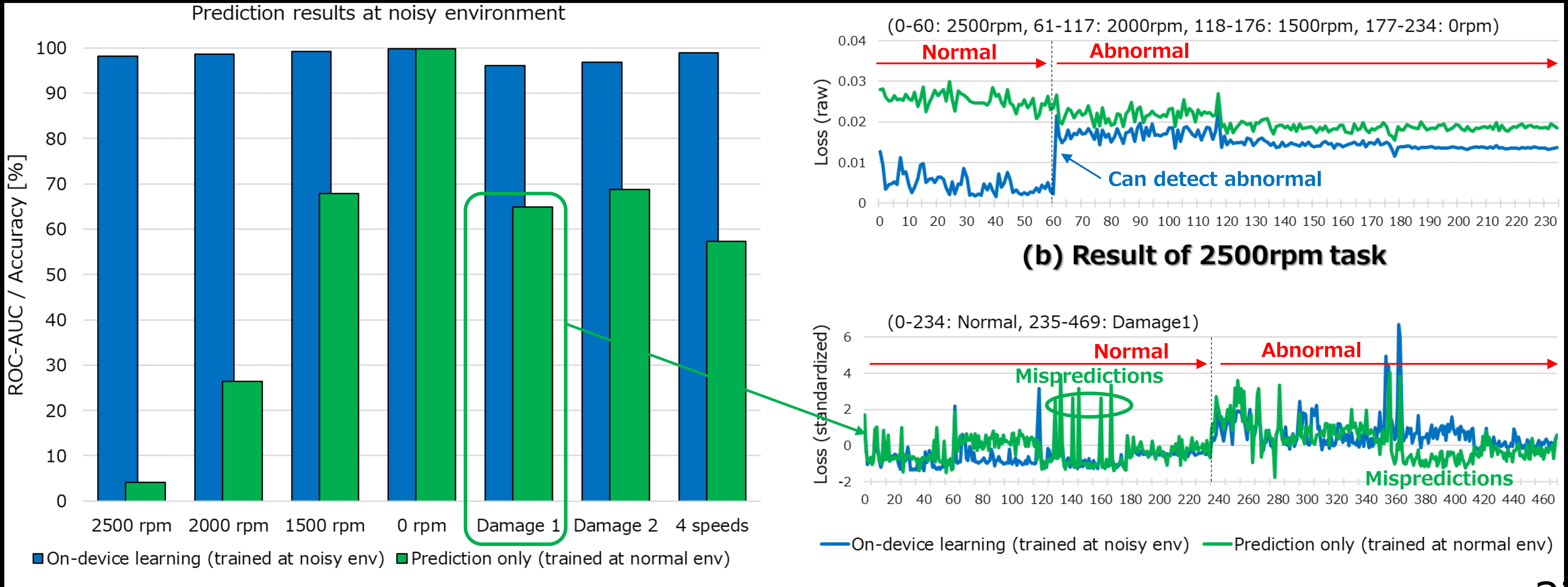
(a) Summary of seven tasks

(b) Result of 2500rpm task

(c) Result of Damage1 task

Evaluations: ROC-AUC / Accuracy

- Trained at normal but predict at noisy environment
  - Predict only: Cannot adapt to the noisy environment ☹
  - On-device learning: Can adapt to the noisy environment ☺

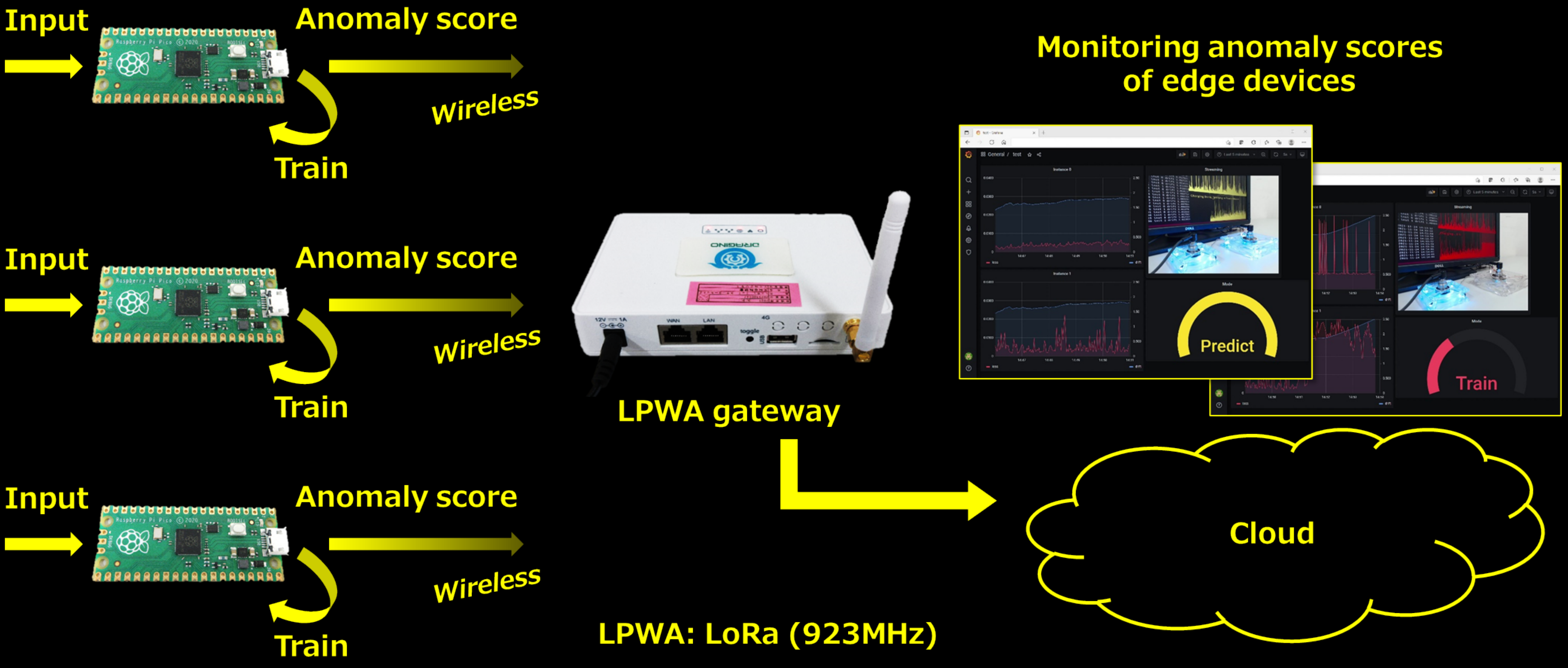


(a) Summary of seven tasks

(b) Result of 2500rpm task

(c) Result of Damage1 task

On-device learning: Summary



IoT sensor	Sensing	Prediction, Training, Monitor & control
Edge AI	Sensing, Prediction	Training, Monitor & control
On-device learning	Sensing, Prediction, Training	Monitor & control