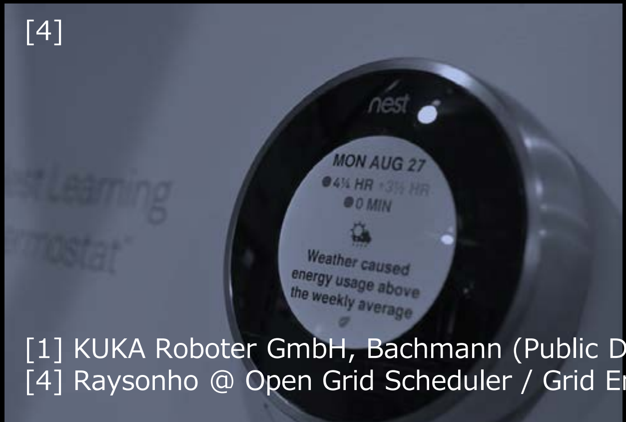
An On-Device Learning Approach for Unsupervised Anomaly Detection

Hiroki Matsutani (Keio University, Japan)
Masaaki Kondo (University of Tokyo, Japan)



[1] KUKA Roboter GmbH, Bachmann (Public Domain) [2] http://www.fatcow.com/data-center-photos [3] Josh Sorenson (Public Domain)
[4] Raysonho @ Open Grid Scheduler / Grid Engine (Public Domain) [5] Sanderflight at Dutch Wikipedia (Public Domain)

Real-world edge AI: Requirements

- Factory, warehouse, robot, HE, security, …

[1] [2] [3]

Real-world anomaly detection

✓ Normal/anomaly patterns vary depending on a given environment and situation

E.g., noise patterns fluctuate

✓ Location of sensors, status of noise sources, …

E.g., normal patterns vary with time

✓ Temperature, workload, human behavior, wear, …

Environment/status are changing

→ Not easy to prepare training data sets beforehand

[4] [5] [6]

[1] KUKA Roboter GmbH, Bachmann (Public Domain) [2] http://www.fatcow.com/data-center-photos [3] Josh Soren
[4] Raysonho @ Open Grid Scheduler / Grid Engine (Public Domain) [5] Sanderflight at Dutch Wikipedia (Public Doma

Our approach: On-device learning

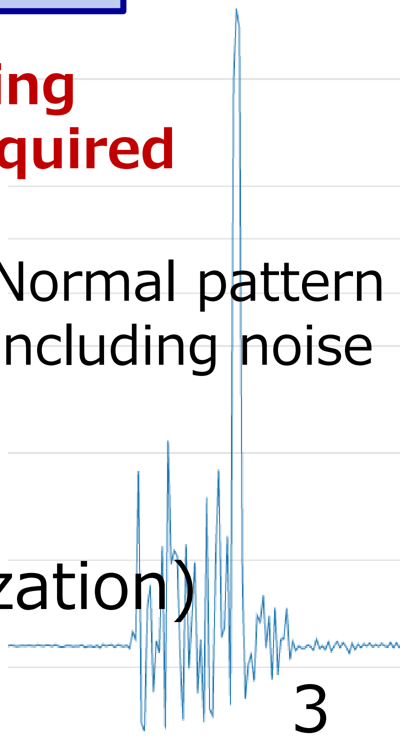
One of the biggest issues when applying AI to industry is to prepare accurate training data sets

<u>On-device learning</u>

Online learning + Unsupervised

On-site learning w/o pre-training phase

Labeled training data is not required



<u>Basic concept</u>

(1) Anomaly detector is deployed
(2) Normal pattern incl. noise is learned (initialization)
(3) Unsupervised anomaly detection

Our approach: On-device learning

Typical DNN based approaches

On-site learning w/o pre-training phase

Offline learning

Online learning

Prediction only

Prediction + Sequential learning

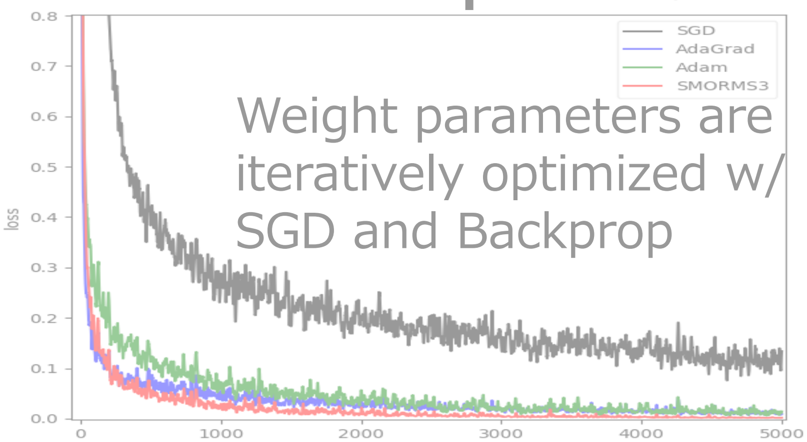
Anomaly detection

Test data

Pre-trained predictor

Learning predictor

Test data ≒ Training data



Weight parameters are iteratively optimized w/ SGD and Backprop

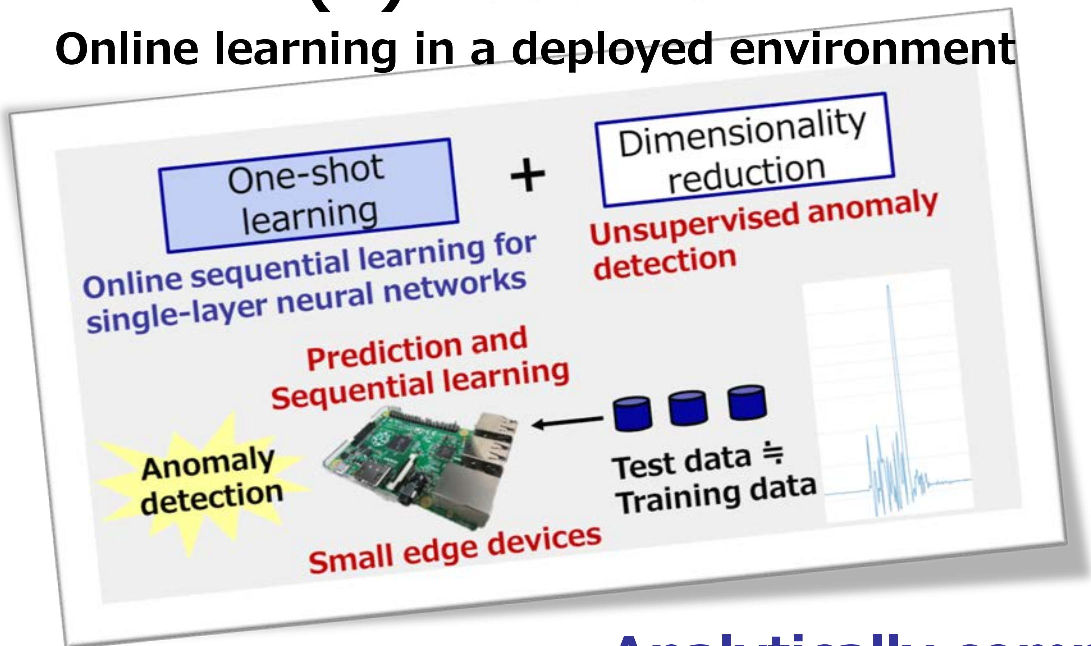
On-device learning for adapting to a given environment

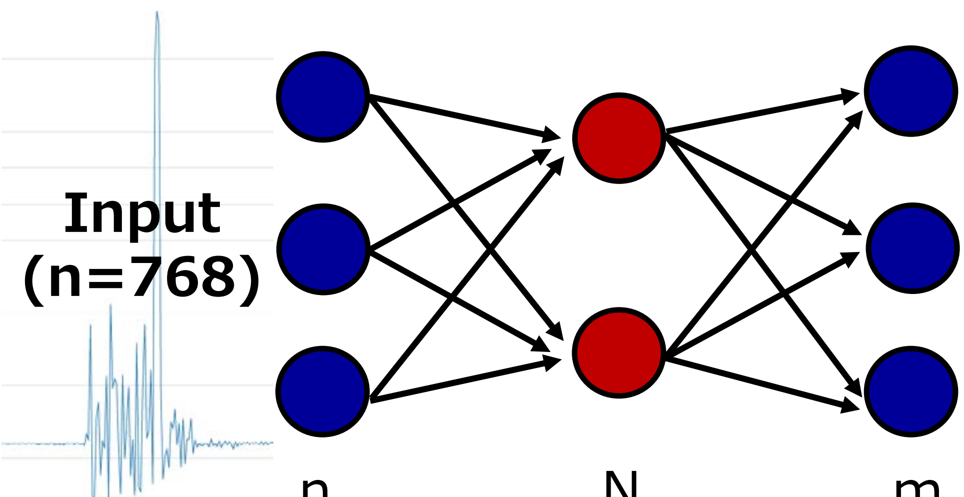
Low cost edge devices

On-device learning: A baseline

(1) Baseline

Online learning in a deployed environment





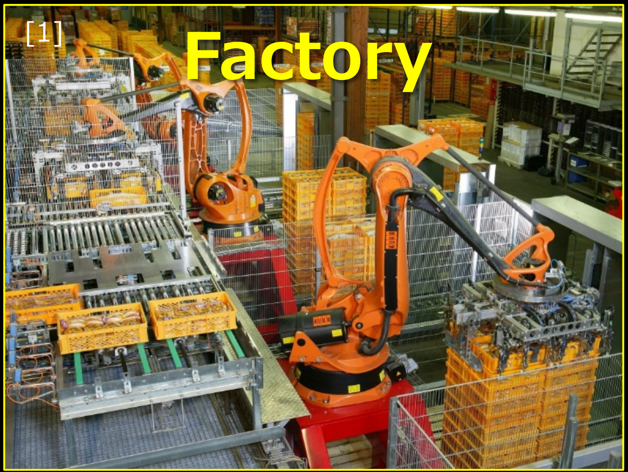
Analytically compute weight parameters w/ "memoization"

E.g., data *i+1* is learned using result of data *i*

Reduced computation cost

Case 1: Manufacture process

- **Finding defects and predictive maintenance**



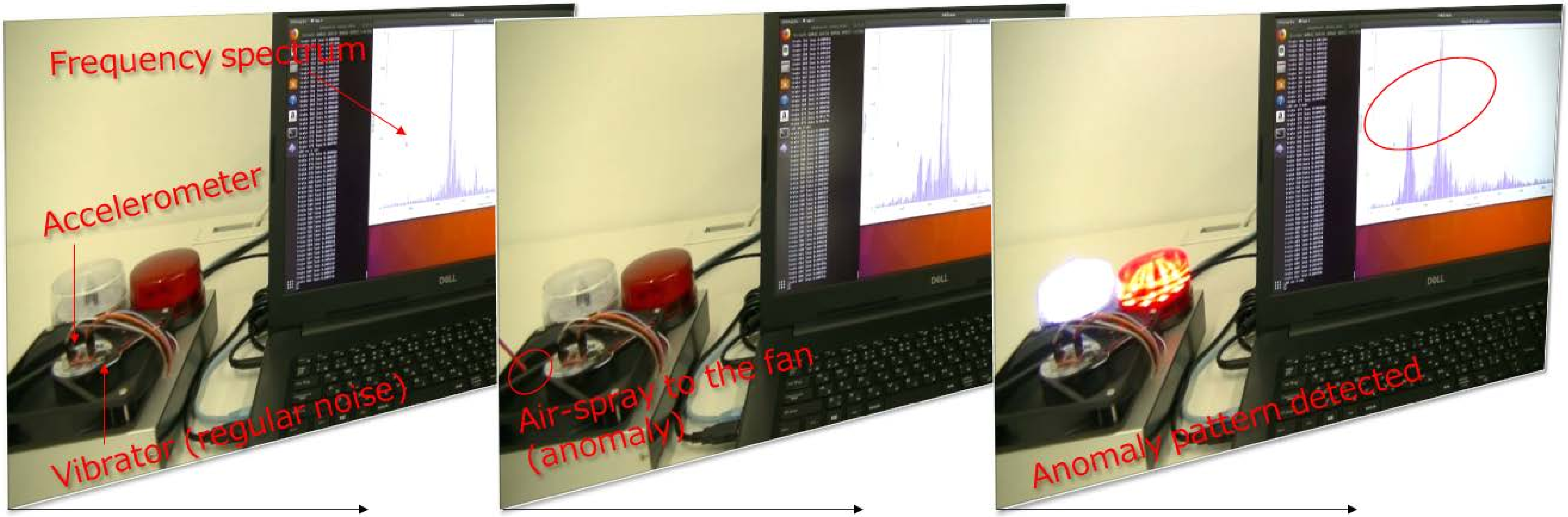
Applied to anomaly detection in manufacture process

- ✓ **Normal samples**
- ✓ **Not-ideal samples**

[1] KUKA Roboter GmbH, Bachmann (Public Domain) [2] http://www.fatcow.com/data-center-photos [3] Josh Sorenson (Public Domain)
[4] Raysonho @ Open Grid Scheduler / Grid Engine (Public Domain) [5] Sanderflight at Dutch Wikipedia (Public Domain)

Case 1: Manufacture process

- **Vibration pattern is learned → Detect unusual event (e.g., air-spray from red tube)**



Step 1:
Normal pattern including regular noise is learned

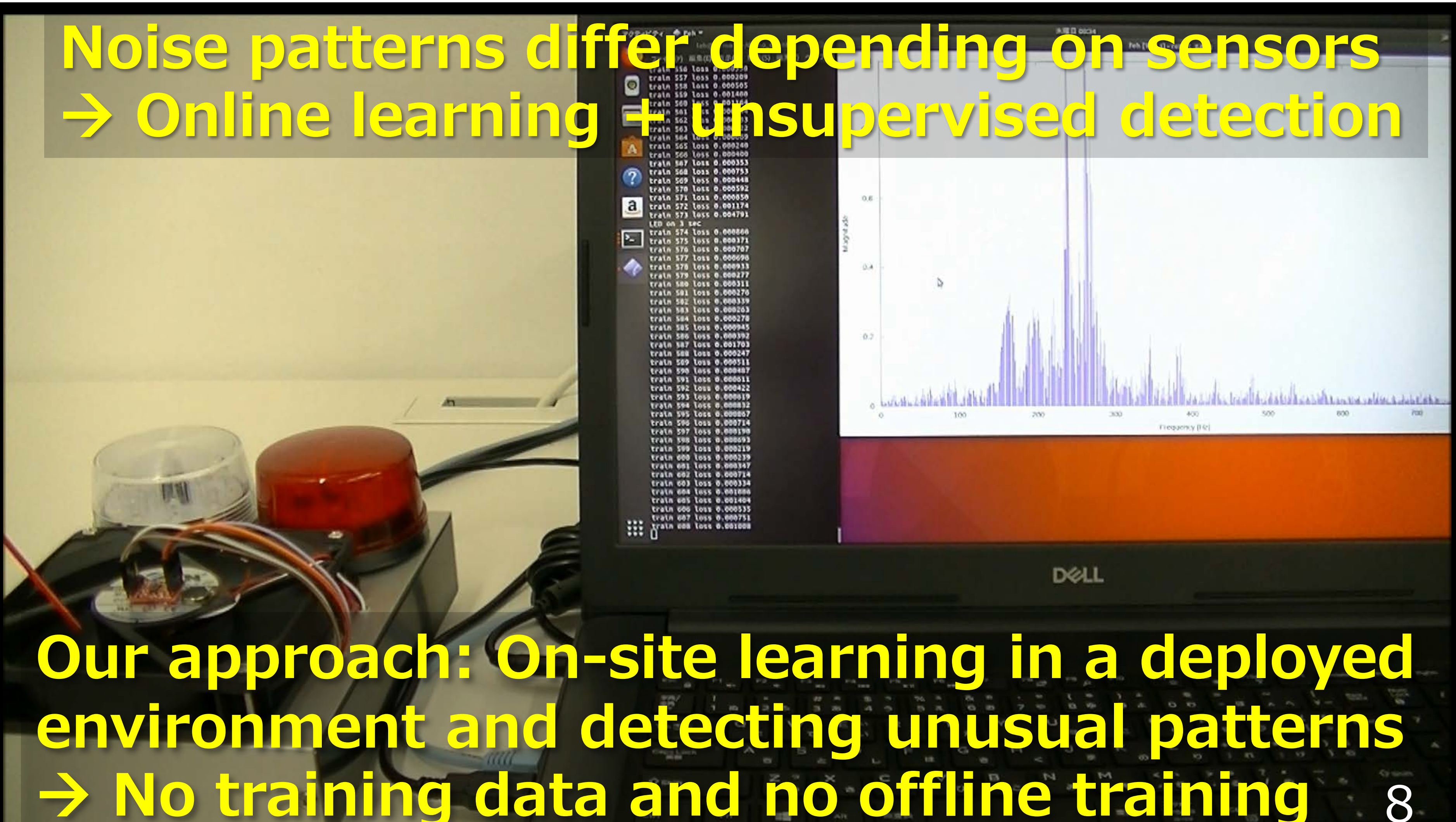
Step 2:
Air-spray is blew to the fan

Step 3:
Anomaly pattern (air-spray) is detected

Our approach: On-site learning in a deployed environment and detecting unusual patterns

Case 1: Manufacture process

- Vibration pattern is learned → Detect unusual event (e.g., air-spray from red tube)



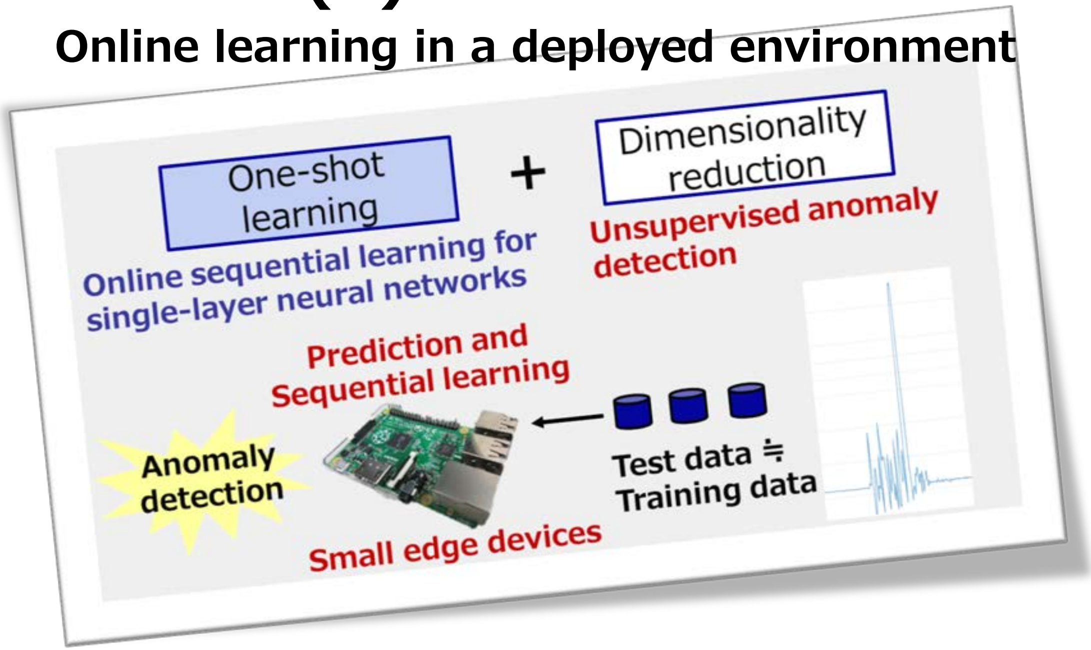
Noise patterns differ depending on sensors
→ Online learning + unsupervised detection

Our approach: On-site learning in a deployed environment and detecting unusual patterns
→ No training data and no offline training

On-device learning: Extensions

(1) Baseline

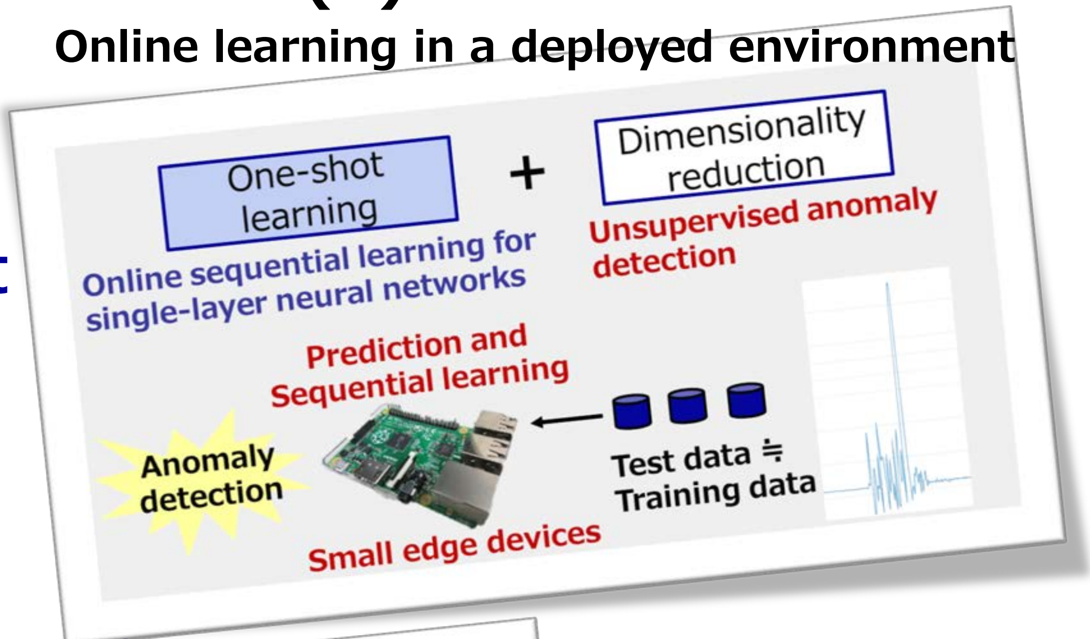
Online learning in a deployed environment



On-device learning: Extensions

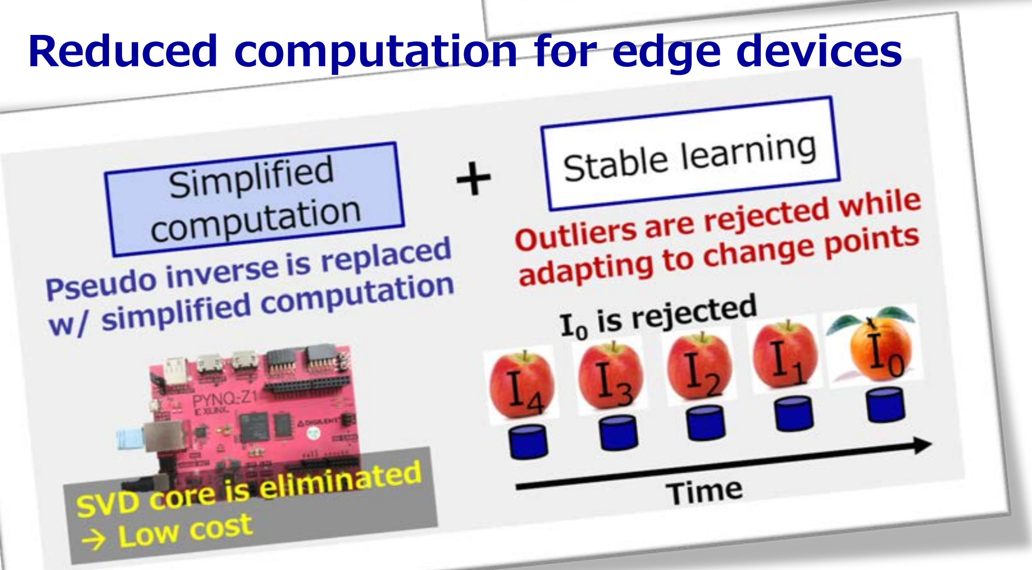
(1) Baseline

Online learning in a deployed environment



(2) Low cost

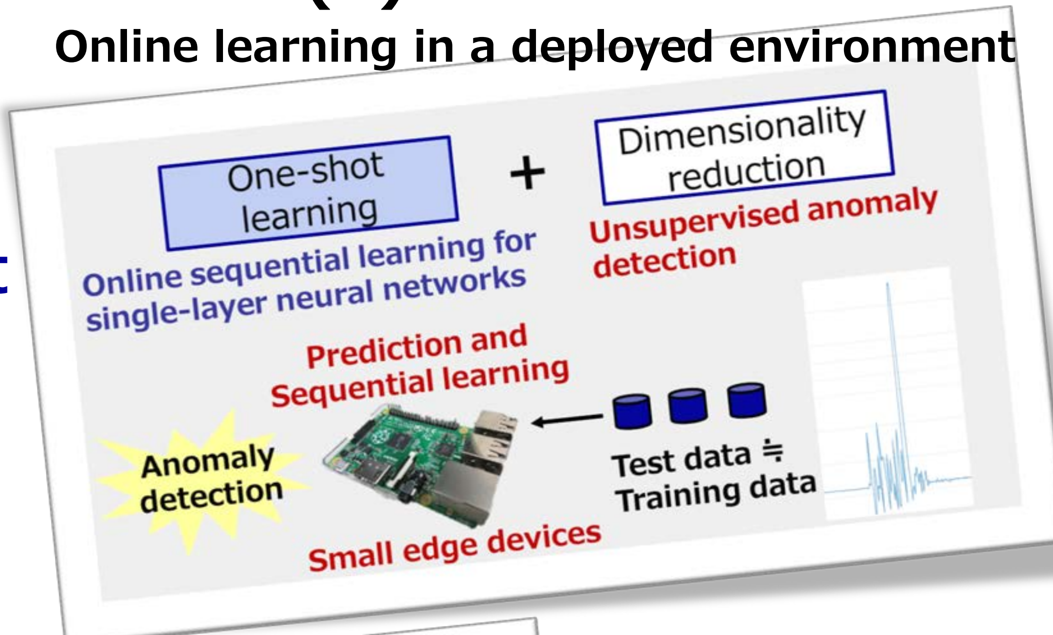
Reduced computation for edge devices



On-device learning: Extensions

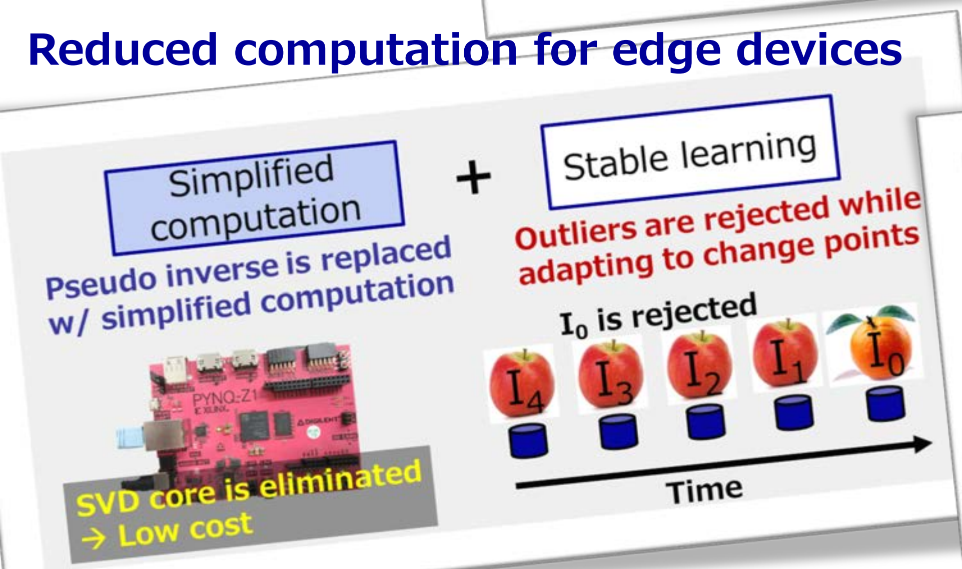
(1) Baseline

Online learning in a deployed environment



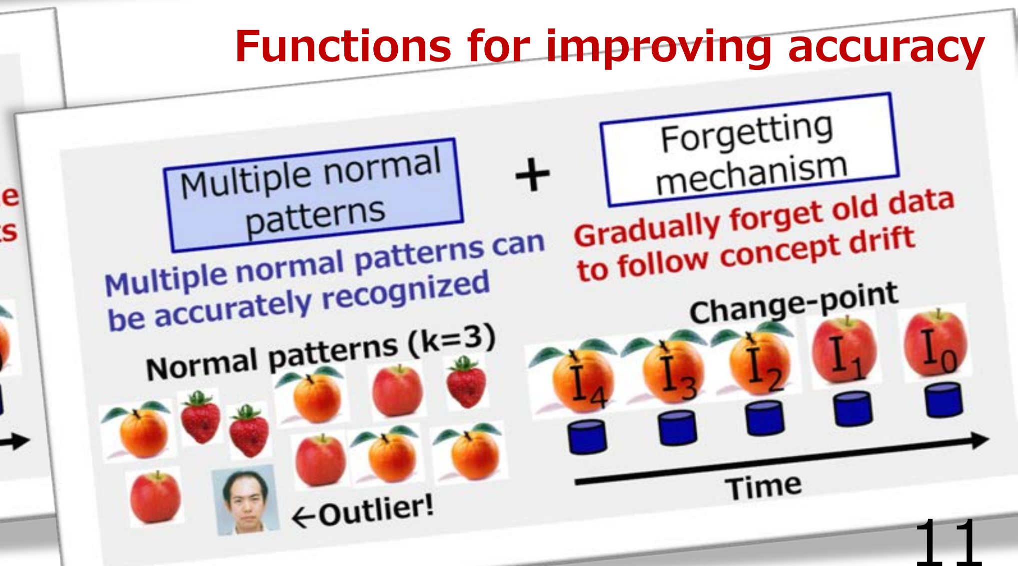
(2) Low cost

Reduced computation for edge devices



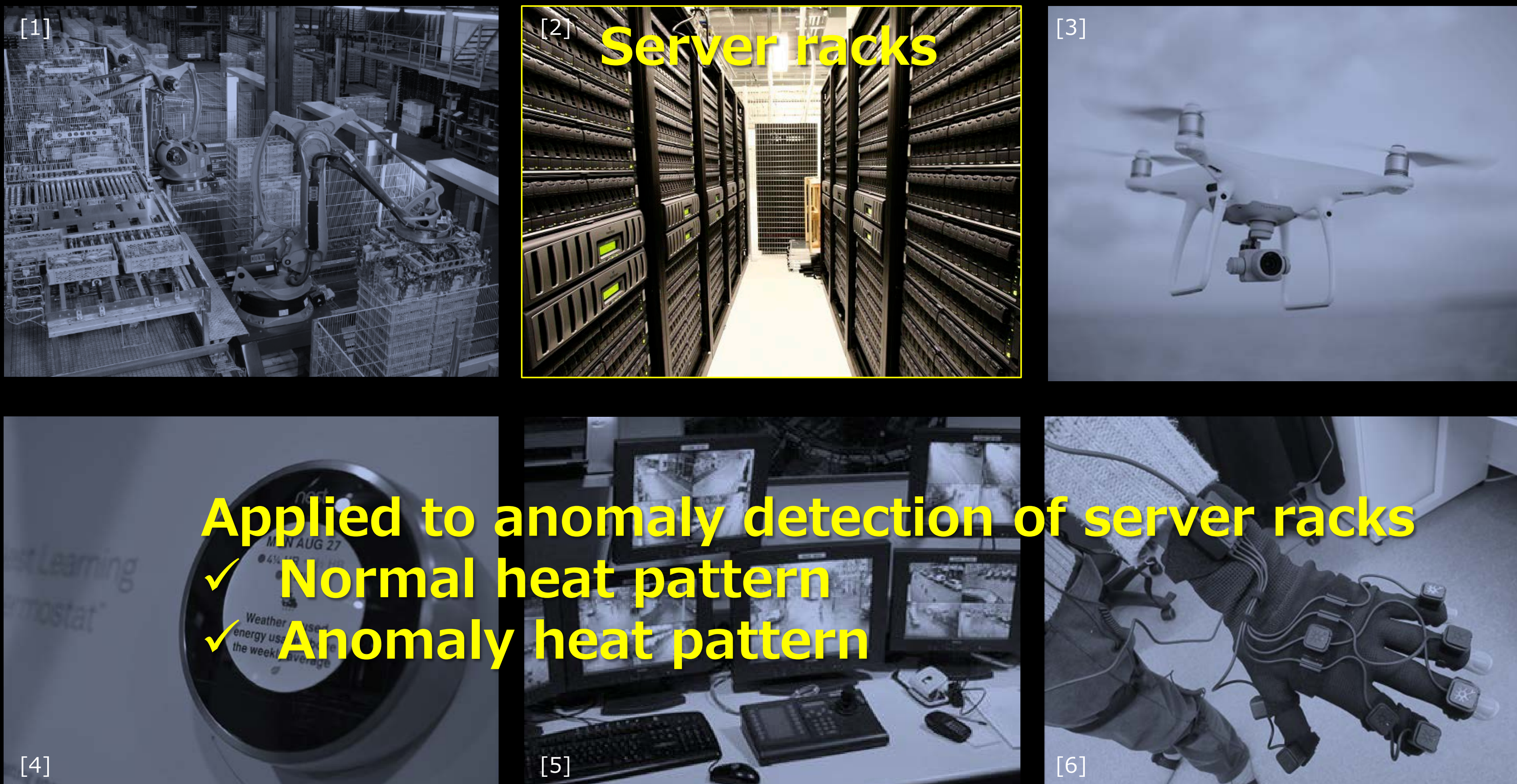
(3) Accuracy

Functions for improving accuracy



Case 2: Server rack & computer

- **Computers and power/cooling components**



Applied to anomaly detection of server racks

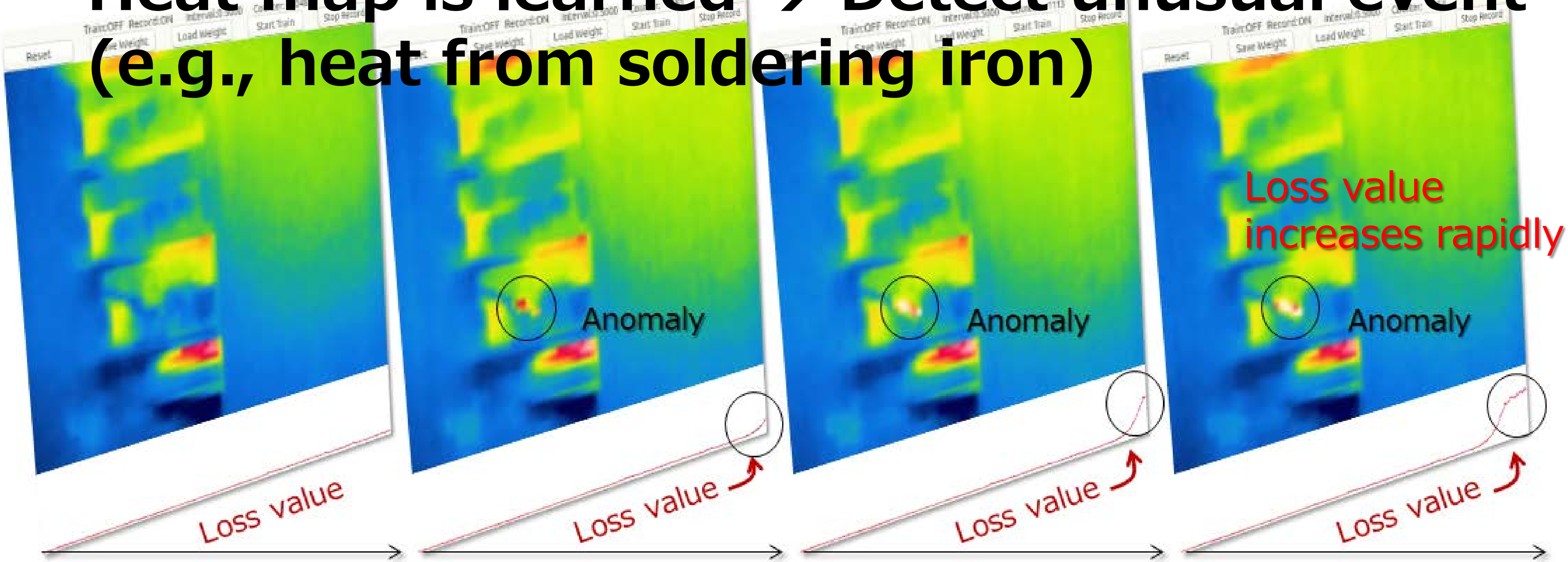
✓ **Normal heat pattern**

✓ **Anomaly heat pattern**

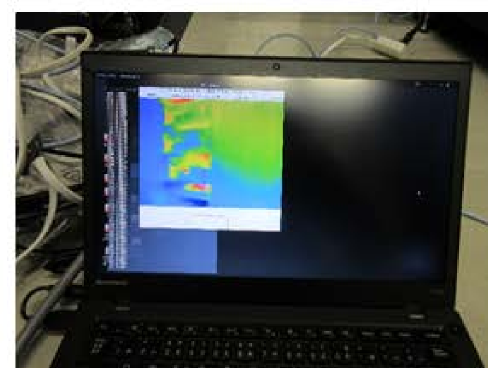
[1] KUKA Roboter GmbH, Bachmann (Public Domain) [2] http://www.fatcow.com/data-center-photos [3] Josh Sorenson (Public Domain)
[4] Raysonho @ Open Grid Scheduler / Grid Engine (Public Domain) [5] Sanderflight at Dutch Wikipedia (Public Domain)

Case 2: Server rack & computer

- **Heat map is learned → Detect unusual event (e.g., heat from soldering iron)**



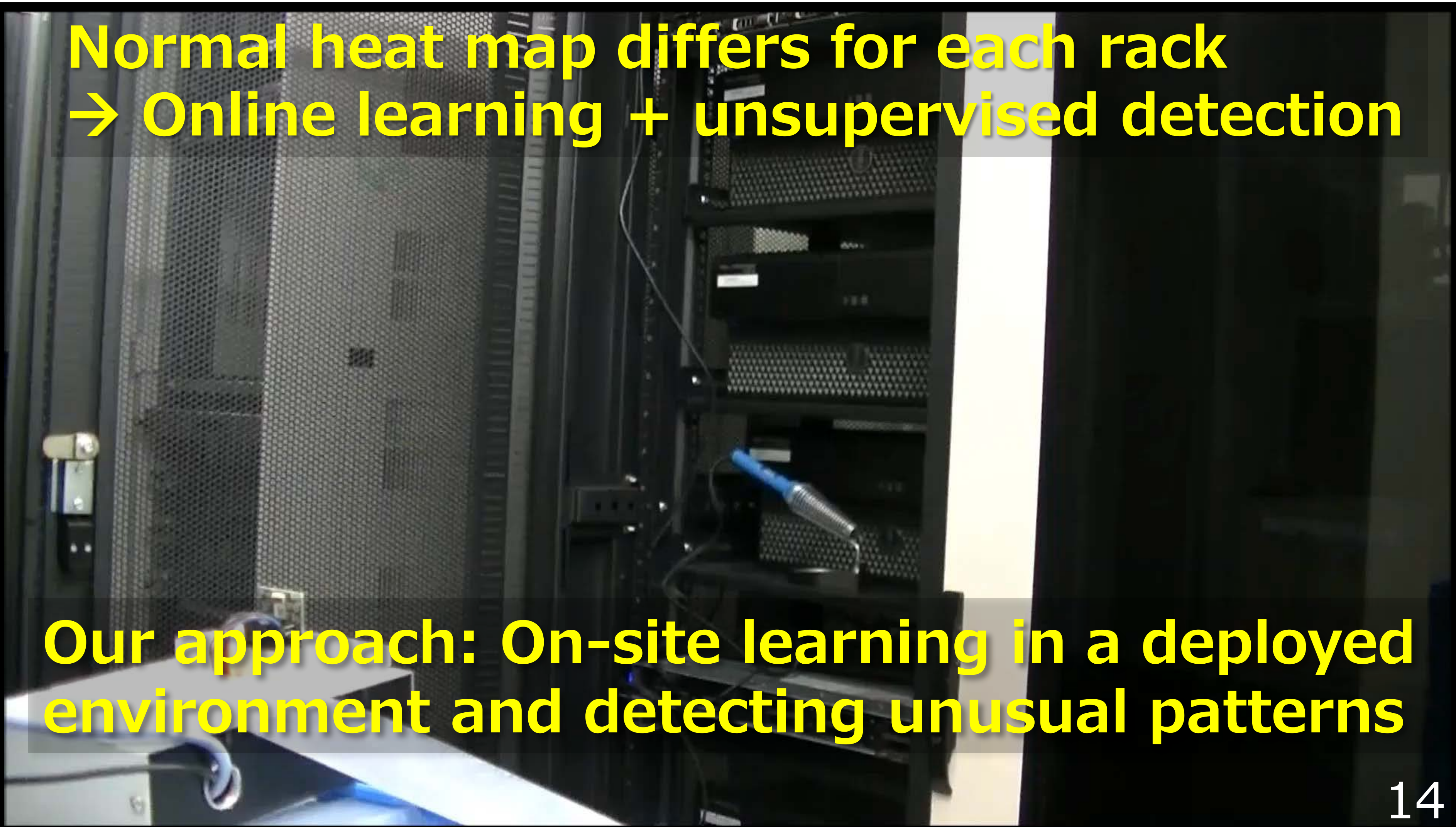
On-device learning camera



Remote monitoring

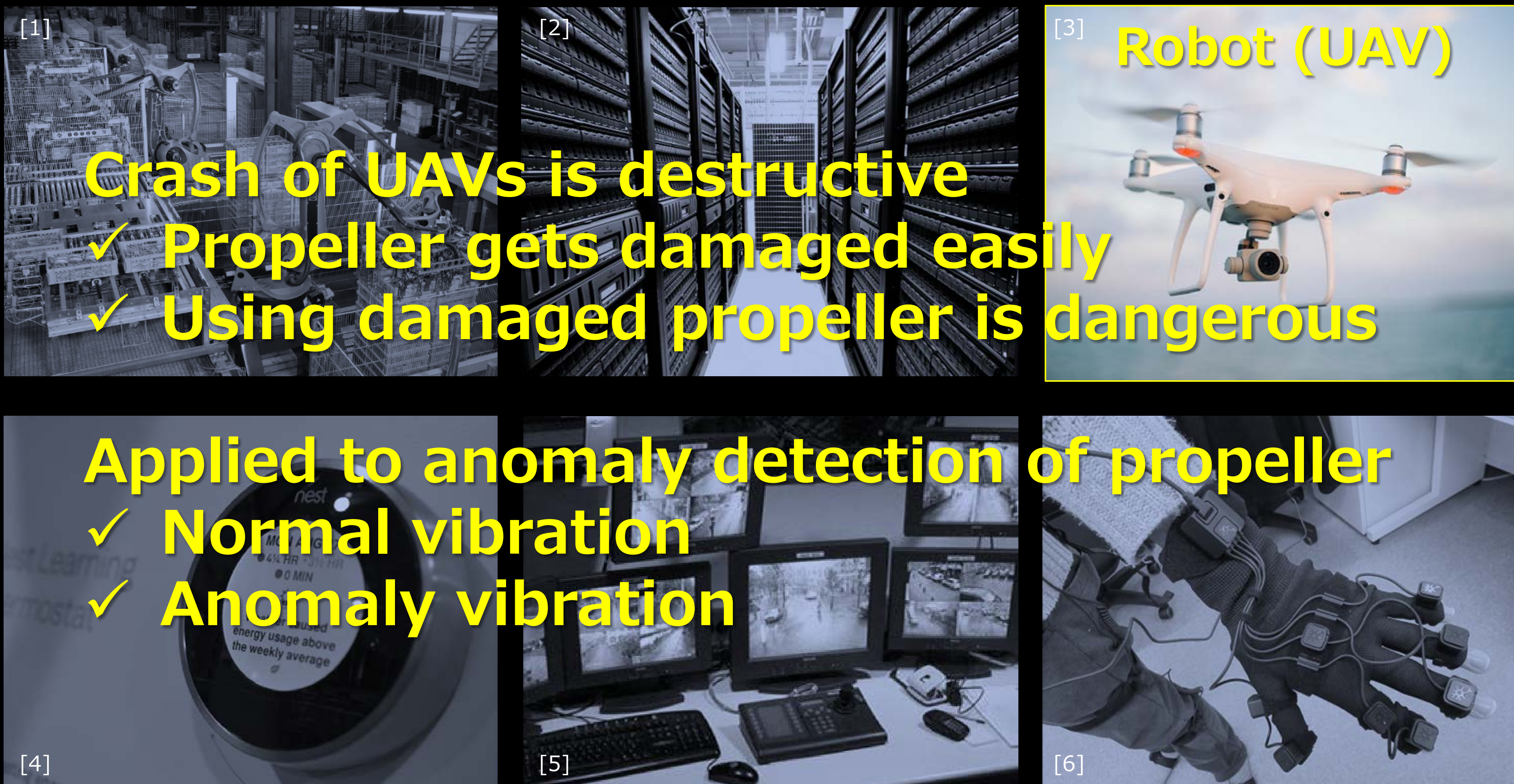
Case 2: Server rack & computer

- Heat map is learned → Detect unusual event (e.g., heat from soldering iron)



Case 3: Mobile robot (UAV)

- **UAV's status depends on payload/condition**



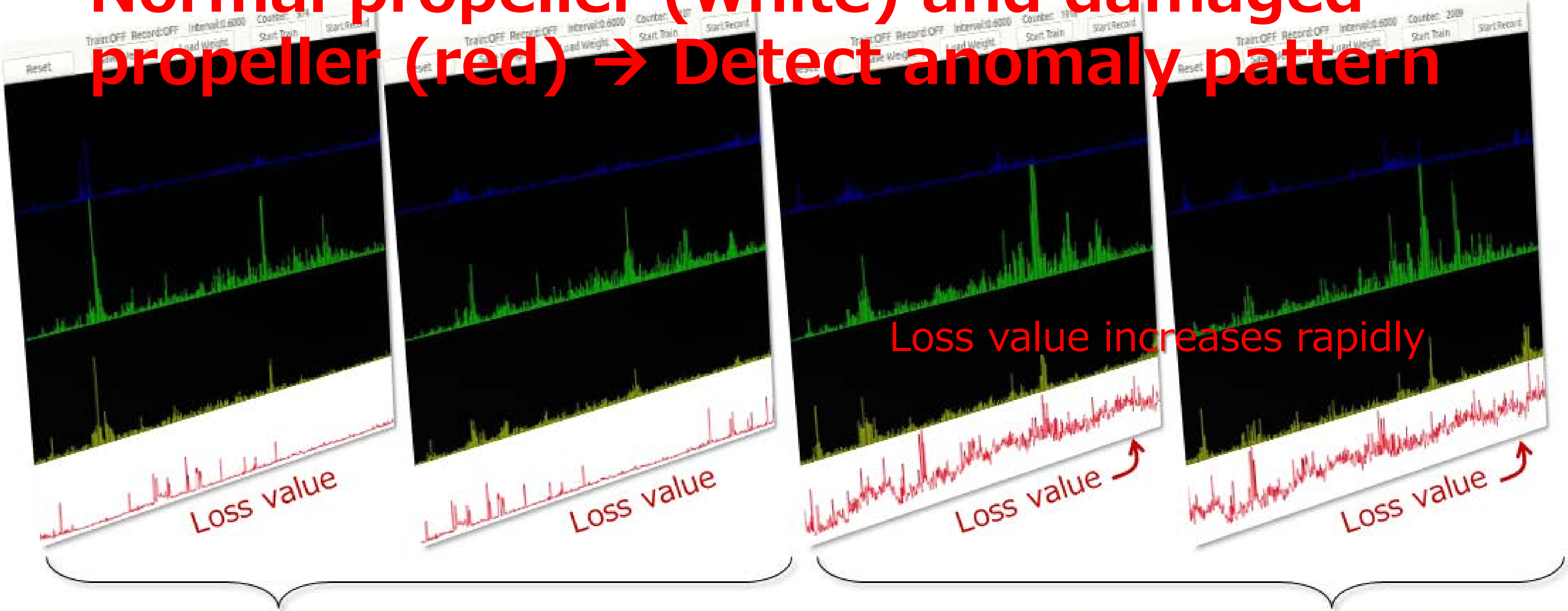
Crash of UAVs is destructive
✓ **Propeller gets damaged easily**
✓ **Using damaged propeller is dangerous**

Applied to anomaly detection of propeller
✓ **Normal vibration**
✓ **Anomaly vibration**

[1] KUKA Roboter GmbH, Bachmann (Public Domain) [2] http://www.fatcow.com/data-center-photos [3] Josh Sorenson (Public Domain)
[4] Raysonho @ Open Grid Scheduler / Grid Engine (Public Domain) [5] Sanderflight at Dutch Wikipedia (Public Domain)

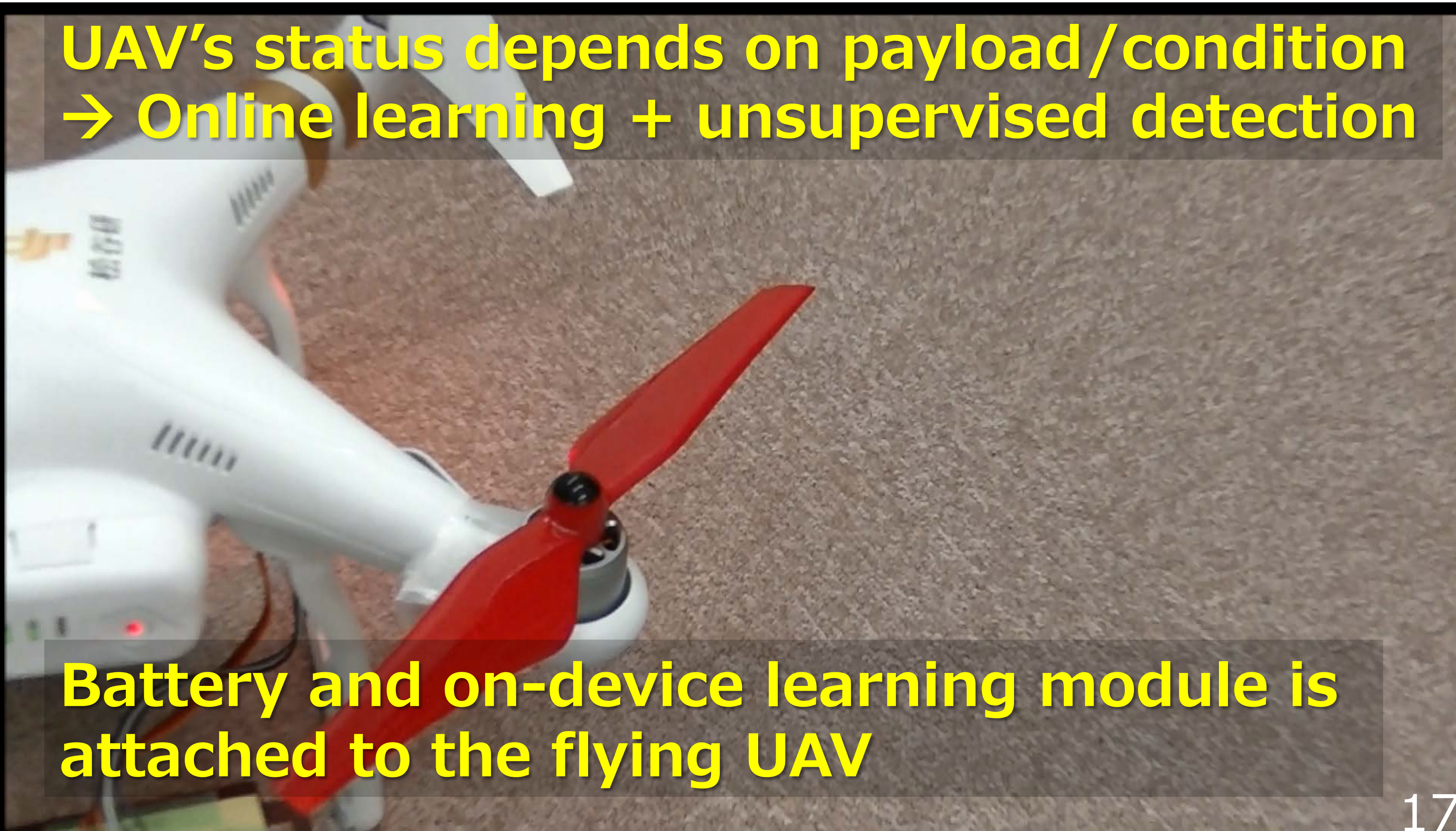
Case 3: Mobile robot (UAV)

- **Normal propeller (white) and damaged propeller (red) → Detect anomaly pattern**



Case 3: Mobile robot (UAV)

- **Normal propeller (white) and damaged propeller (red) → Detect anomaly pattern**



On-device learning: Summary

One of the biggest issues when applying AI to industry is to prepare accurate training data sets

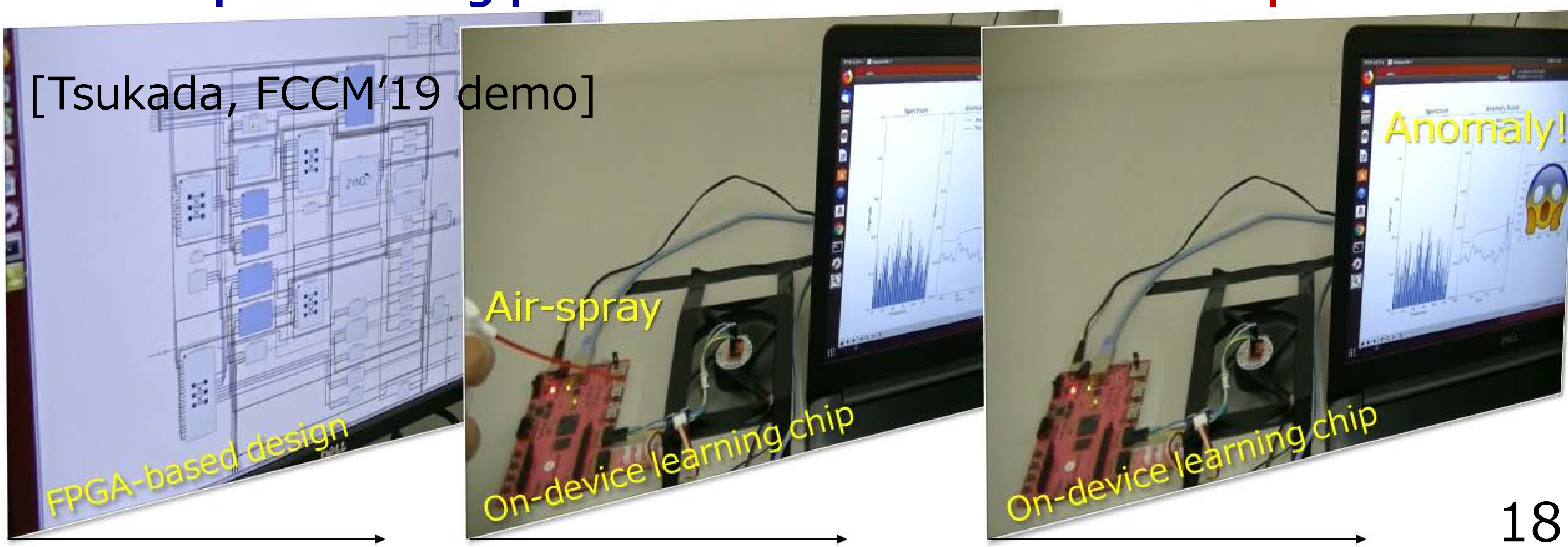
<u>On-device learning</u>

Online learning + Unsupervised

On-site learning w/o pre-training phase

Labeled training data is not required

[Tsukada, FCCM'19 demo]



References (1/2)

- On-device learning anomaly detection
  - Mineto Tsukada, et al., "A Neural Network Based On-Device Learning Anomaly Detector for Edge Devices", arXiv:1907.10147 (2019).
  - Mineto Tsukada, et al., "An FPGA-based On-device Sequential Learning Approach for Unsupervised Anomaly Detection", FCCM 2019 Demo Night.
  - Mineto Tsukada, et al., "OS-ELM-FPGA: An FPGA-Based Online Sequential Unsupervised Anomaly Detector", Euro-Par Workshops 2018.

References (2/2)

- On-device learning core
  - Tomoya Itsubo, et al., "Performance and Cost Evaluations of Online Sequential Learning and Unsupervised Anomaly Detection Core", IEEE COOL Chips 2019.
- Abnormal behavior detection
  - Rei Ito, et al., "An Adaptive Abnormal Behavior Detection using Online Sequential Learning", IEEE EUC 2019.