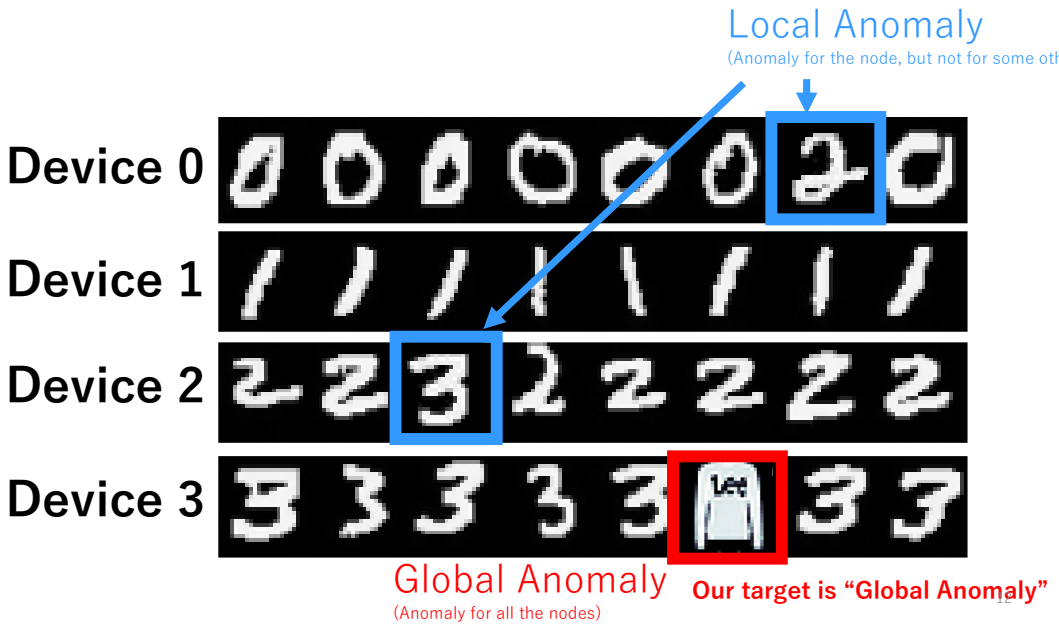
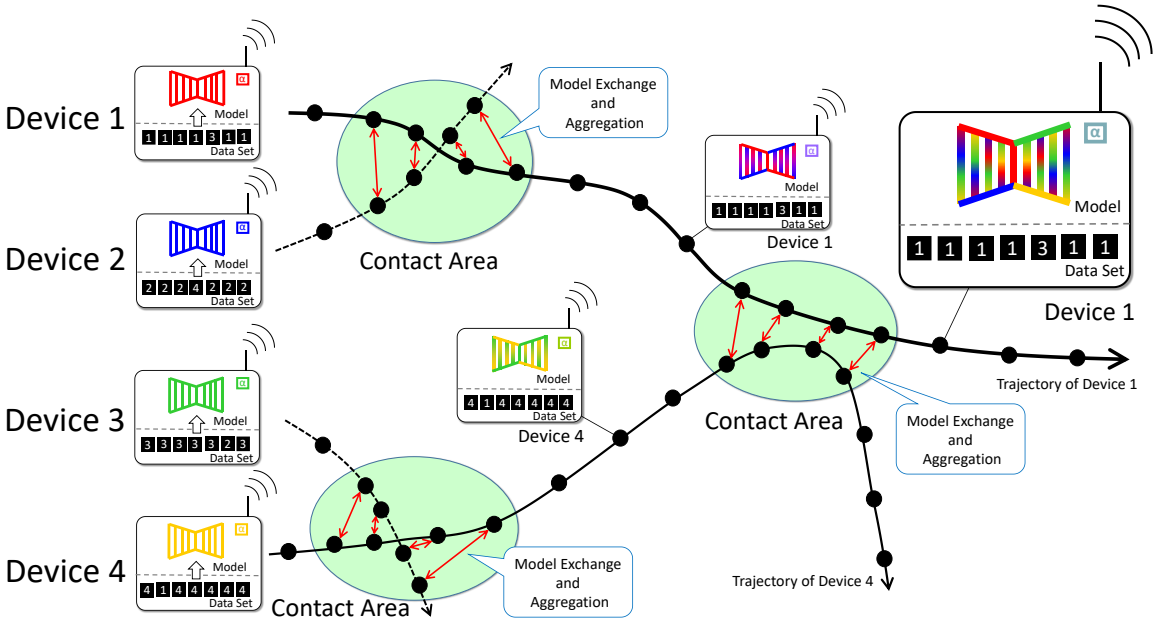
Detection of Global Anomalies on Distributed IoT Edges with Device-to-Device Communication



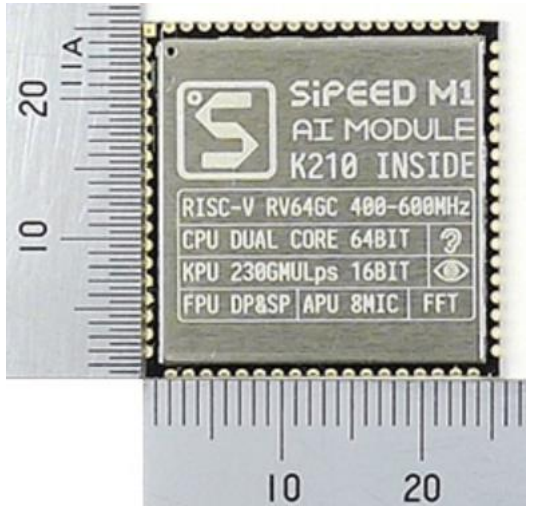


Hideya Ochiai, Riku Nishihata,
Eisuke Tomiyama, Yuwei Sun, Hiroshi Esaki
The University of Tokyo, Japan

Table of Contents

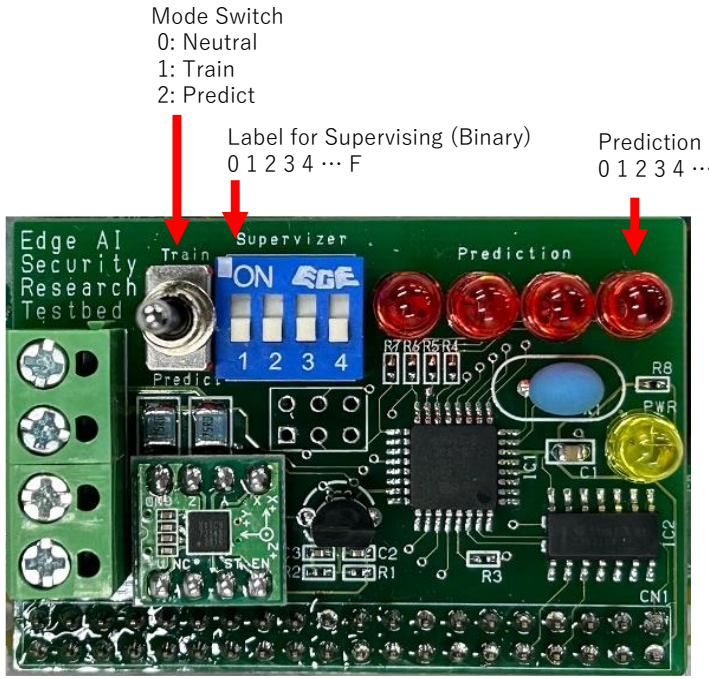
Toward On-Device Training

- Machine Learning has evolved with cloud computing.
- Nowadays :
  - AI Chips are available for 10 USD.
  - Machine Learning shifts onto IoT Edges.



AI Chip (10USD)

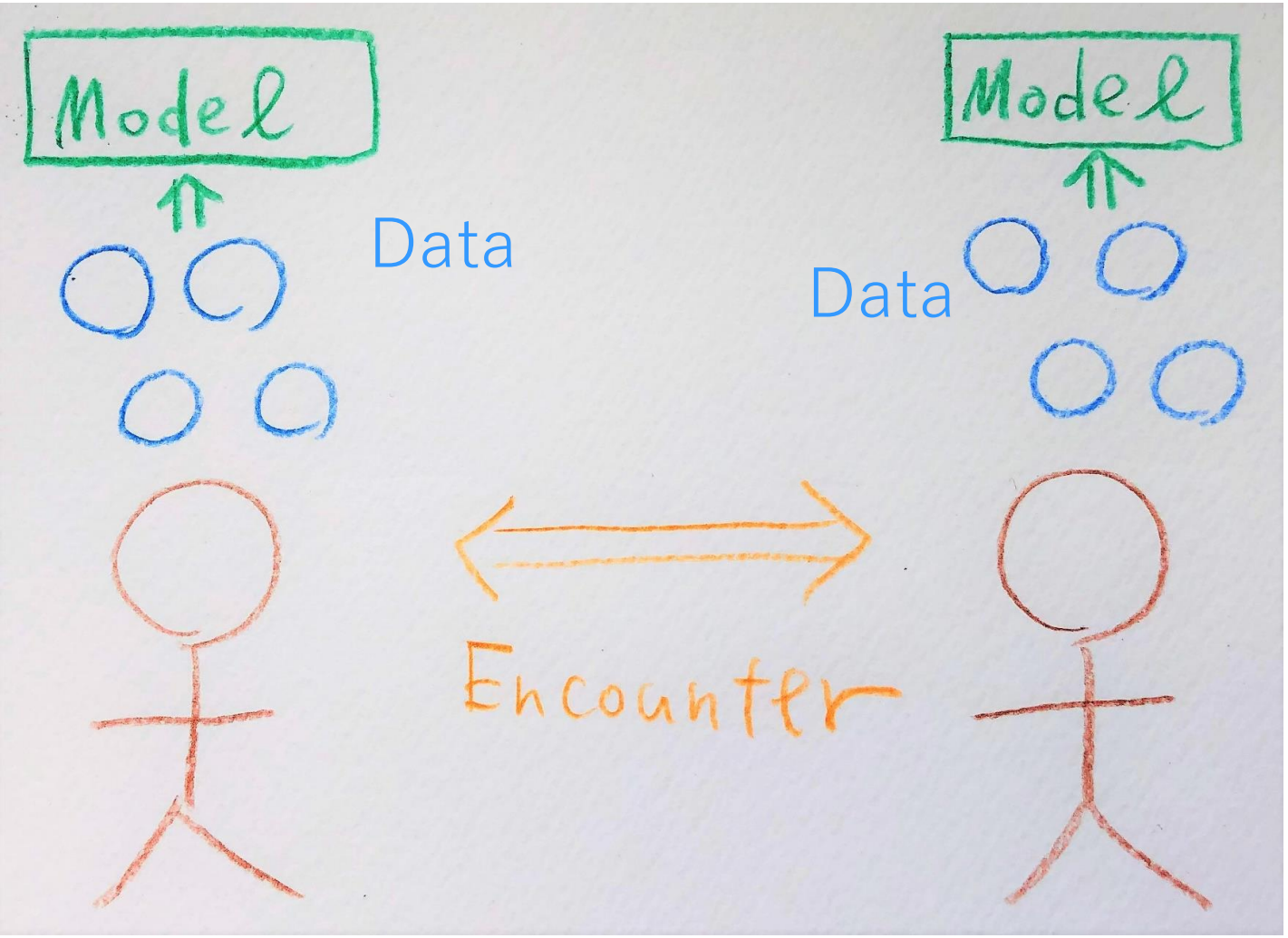
- Issues:
  - Learning on an Edge may lead to model overfit.
- Approach of this issue:
  - Collaborative Training with Device-to-Device Communication.
- Focus of this research:
  - Training Distributed Autoencoder
  - Global Anomaly Detection with Benchmark Settings



Edge AI Testbed

Table of Contents

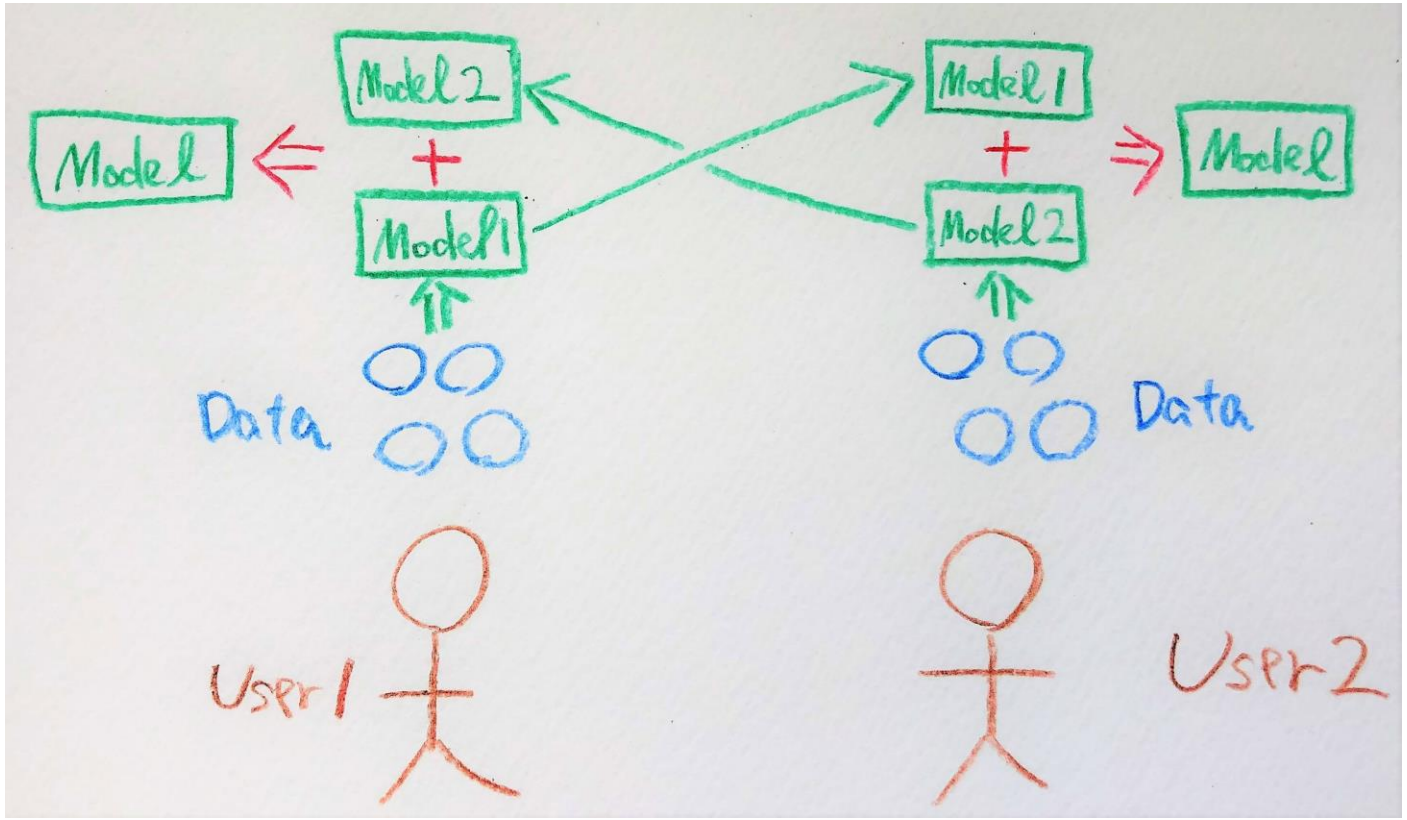
Previous Studies (1/5): Wireless Ad Hoc Federated Learning (WAFL) [1] with Device-to-Device Communication



1. Each node individually trains its ML model using its local data.
2. Each node encounters the other.
3. They can communicate with local wireless communication media such as Wi-Fi Ad Hoc mode or Bluetooth

[1] Ochiai, Hideya, et al. "Wireless ad hoc federated learning: A fully distributed cooperative machine learning." *arXiv preprint arXiv:2205.11779* (2022).

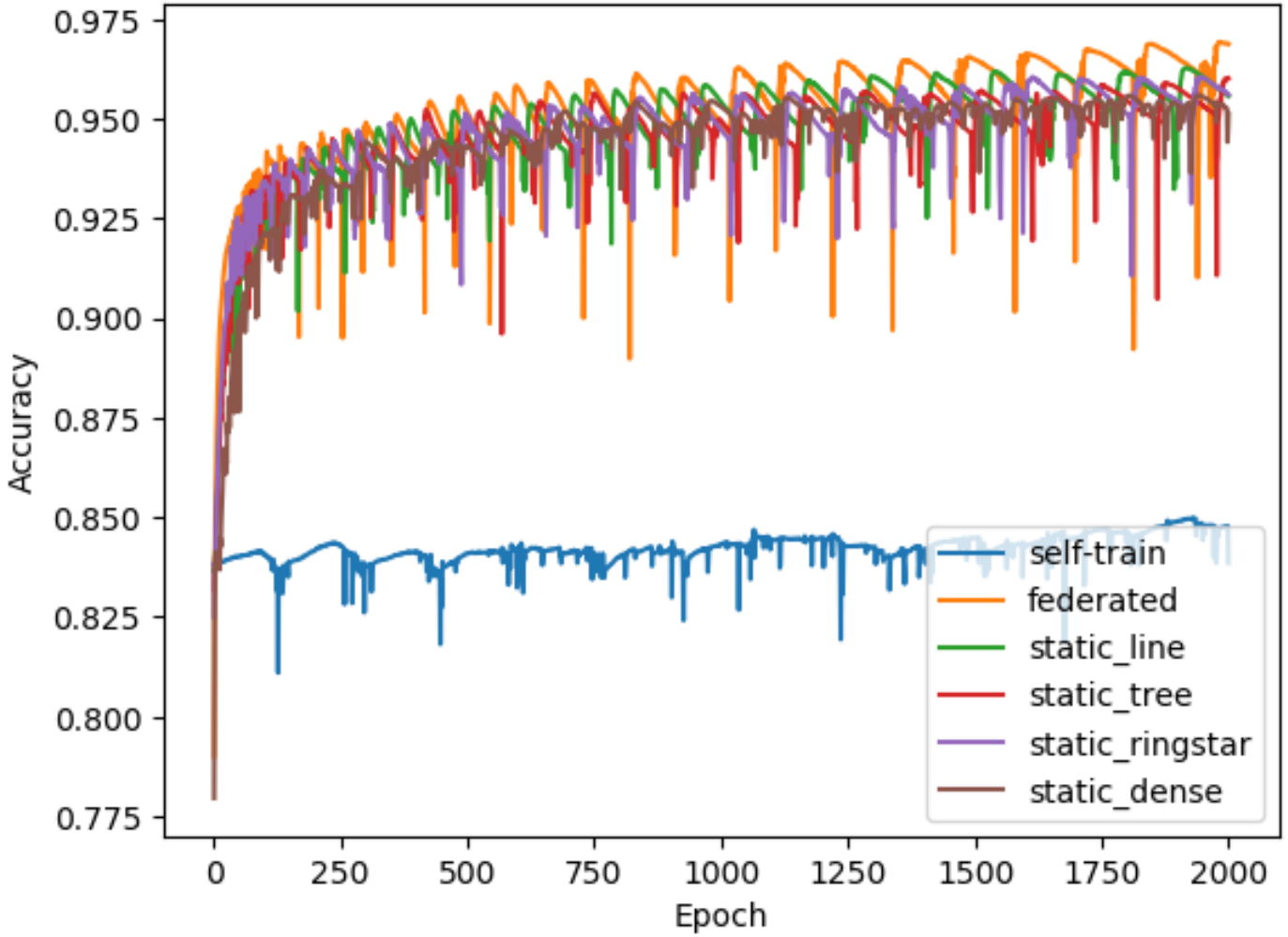
Previous Studies (2/5): Wireless Ad Hoc Federated Learning (WAFL) [1] with Device-to-Device Communication



1. Each node individually trains its ML model using its local data.
2. Each node encounters the other.
3. They can communicate with local wireless communication media such as Wi-Fi Ad Hoc mode or Bluetooth
4. They exchange and aggregate the models to develop a new model.
5. This enables collaborative training.

[1] Ochiai, Hideya, et al. "Wireless ad hoc federated learning: A fully distributed cooperative machine learning." *arXiv preprint arXiv:2205.11779* (2022).

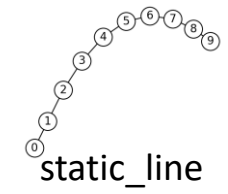
Previous Studies (3/5): Wireless Ad Hoc Federated Learning (WAFL) [1] with Device-to-Device Communication

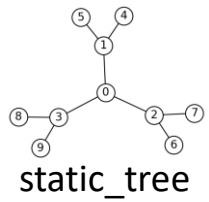


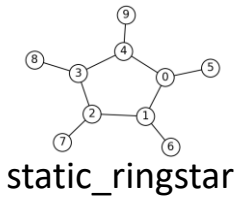
WAFL has achieved almost the same performance as conventional federated learning.

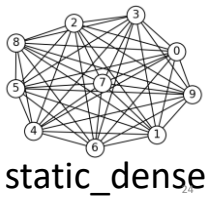
10.0-11.5% Improvement

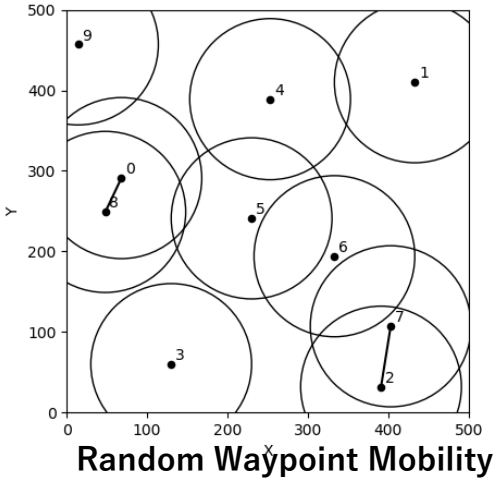
Accuracy of Self-Train







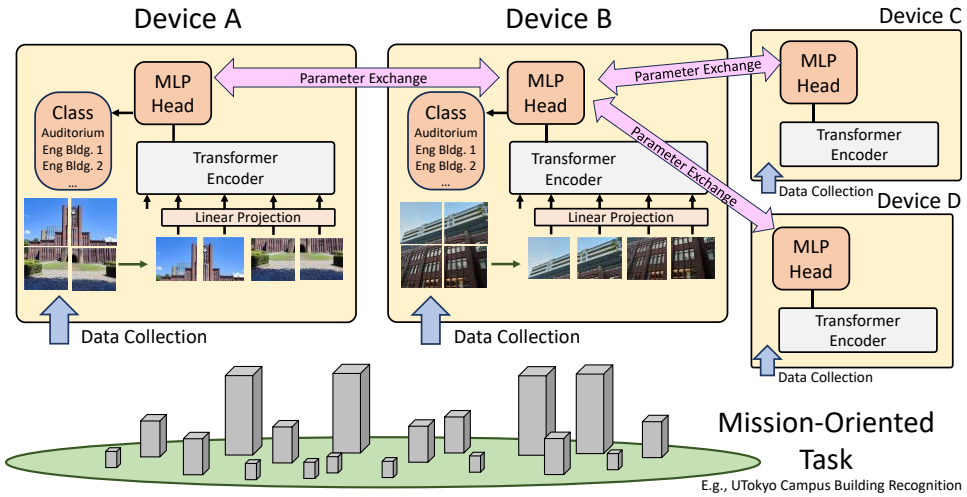




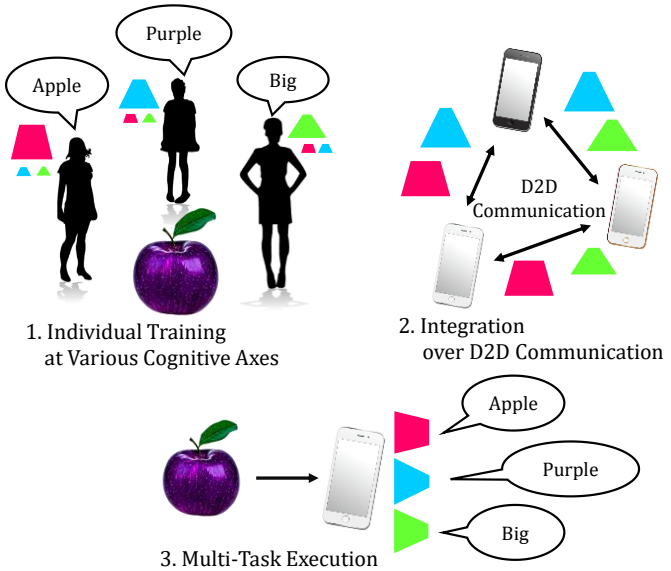
Ochiai, Hideya, et al. "Wireless ad hoc federated learning: A fully distributed cooperative machine learning." *arXiv preprint arXiv:2205.11779* (2022).

Previous Studies (4/5): Extensions and Variations of WAFL (1/2)

WAFL-Vision Transformer [1]



WAFL- Multi-Task Learning[2]



WAFL-GAN[3]

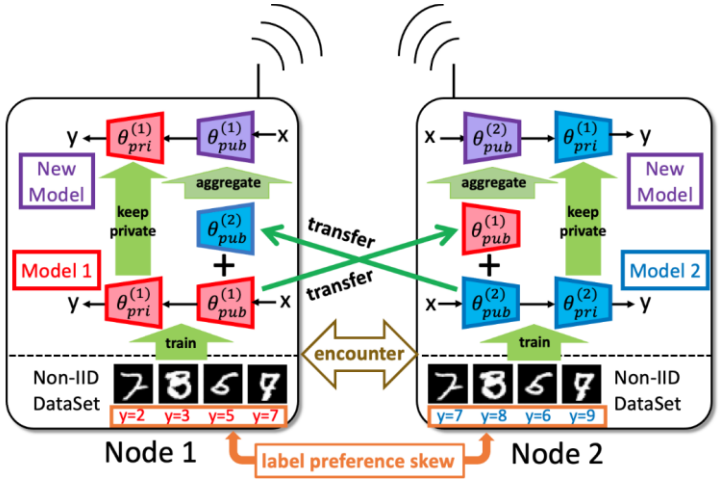
[1] **Hideya Ochiai, Atsuya Muramatsu, Yudai Ueda, Ryuhei Yamaguchi, Kazuhiro Katoh, and Hiroshi Esaki, "Tuning Vision Transformer with Device-to-Device Communication for Targeted Image Recognition", IEEE World Forum on Internet of Things, 2023 (Best Paper Award).**

[2] Ryusei Higuchi, Hiroshi Esaki, Hideya Ochiai, "Collaborative Multi-Task Learning across Internet Edges with Device-to-Device Communications“, IEEE Cybermatics Congress, 2023 (under review).

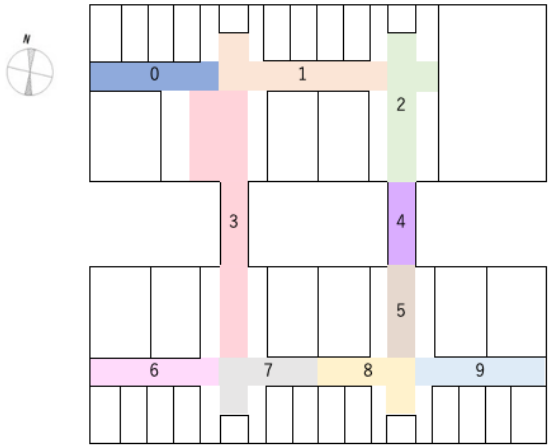
[3] Eisuke Tomiyama, Hiroshi Esaki, Hideya Ochiai, "WAFL-GAN: Wireless Ad Hoc Federated Learning for Distributed Generative Adversarial Networks", IEEE International Conference on Knowledge and Smart Technology, 2023.

Previous Studies (5/5): Extensions and Variations of WAFL (2/2)

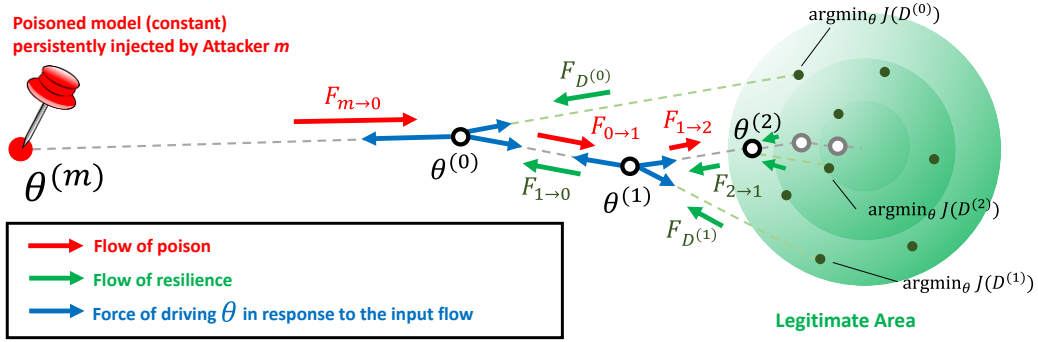
WAFL-Personalization[4]



WAFL-Localization[5]



WAFL-Security[6]



[4] Ryusei Higuchi, Hiroshi Esaki, and Hideya Ochiai, "Personalized Wireless Ad Hoc Federated Learning for Label Preference Skew", IEEE World Forum on Internet of Things, 2023.

[5] Yusuke Sugizaki, Hideya Ochiai, Muhammad Asad, Manabu Tsukada, and Hiroshi Esaki, "Wireless Ad-Hoc Federated Learning for Cooperative Map Creation and Localization Models", IEEE World Forum on Internet of Things, 2023.

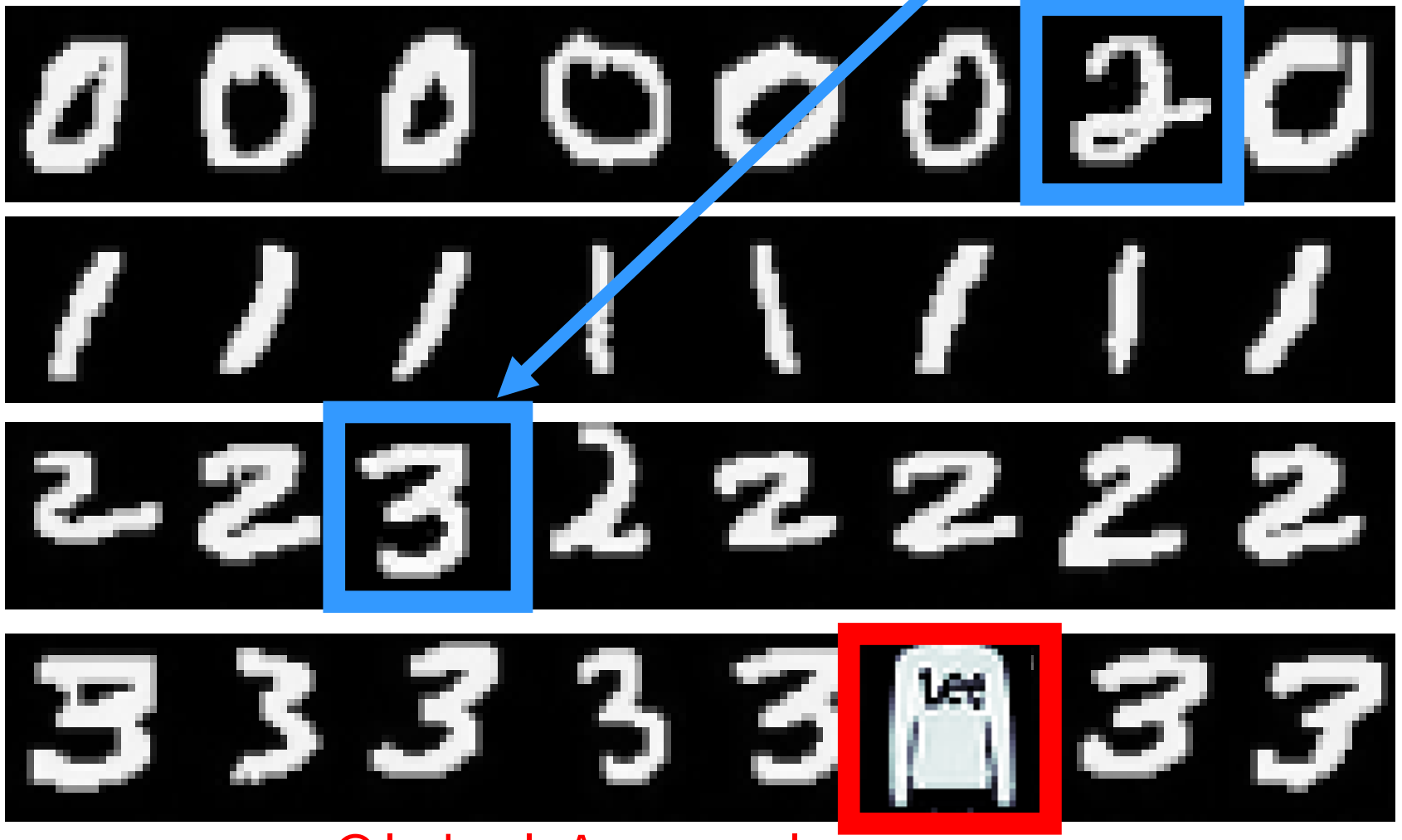
[6] Naoya Tezuka, Hideya Ochiai, Yuwei Sun, Hiroshi Esaki, "Resilience of Wireless Ad Hoc Federated Learning against Model Poisoning Attacks", IEEE International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA), 2022.

For IoT applications, we propose WAFL-Autoencoder for anomaly detection.

Table of Contents

Anomaly Detection in Non-IID Scenario

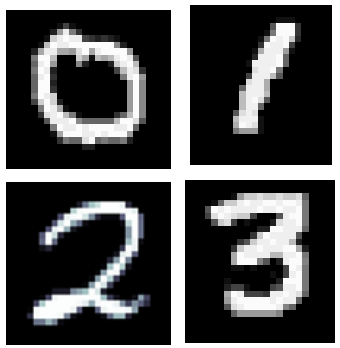
Local Anomaly
(Anomaly for the node, but not for some others)



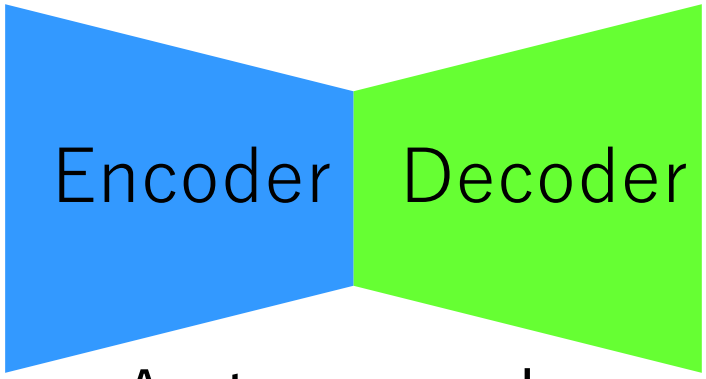
Global Anomaly
(Anomaly for all the nodes)

Our target is “Global Anomaly”

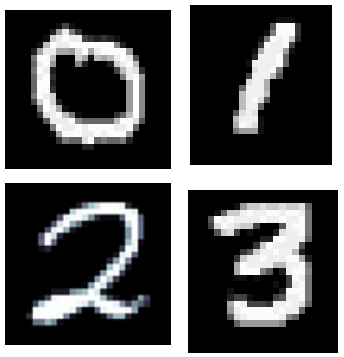
To Detect a Global Anomaly



Legitimate Input



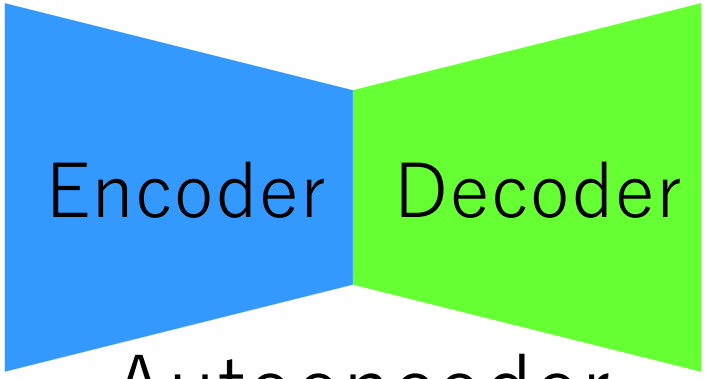
Autoencoder (@ Device n)



Output (Reconstruction)

"To precisely reconstruct the legitimate inputs", WAFL-Autoencoder should learn the legitimate features.

Anomaly Input

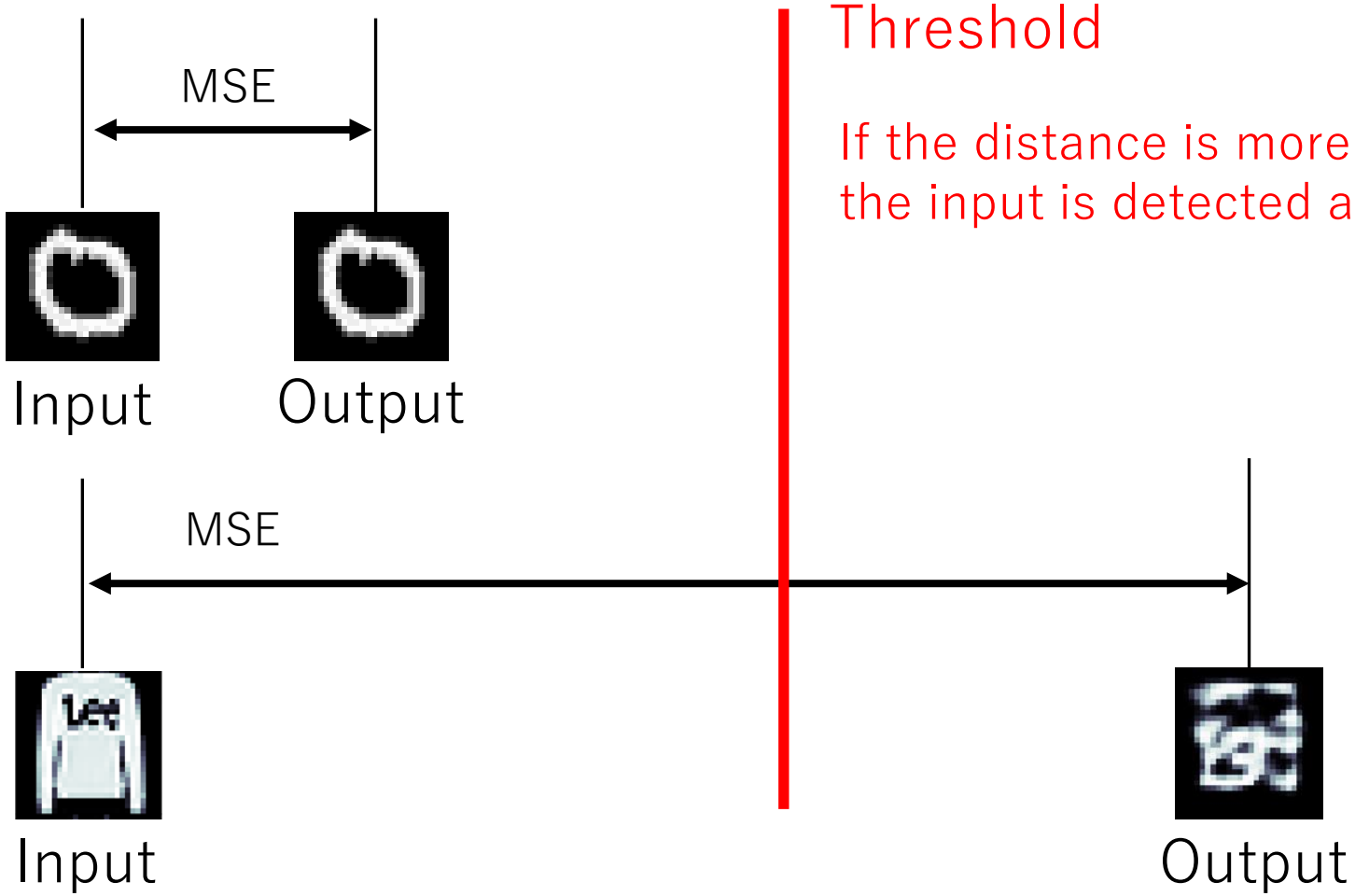


Autoencoder (@ Device n)

Output (Reconstruction)

WAFL-Autoencoder should fail in reconstructing a Global Anomaly input.

Anomaly Detection Thresholds



Threshold

If the distance is more than a calculated threshold, the input is detected as an anomaly!!

Threshold can be calculated locally at the device with its local validation data.

Each device may have different thresholds !! (especially in Non-IID case)

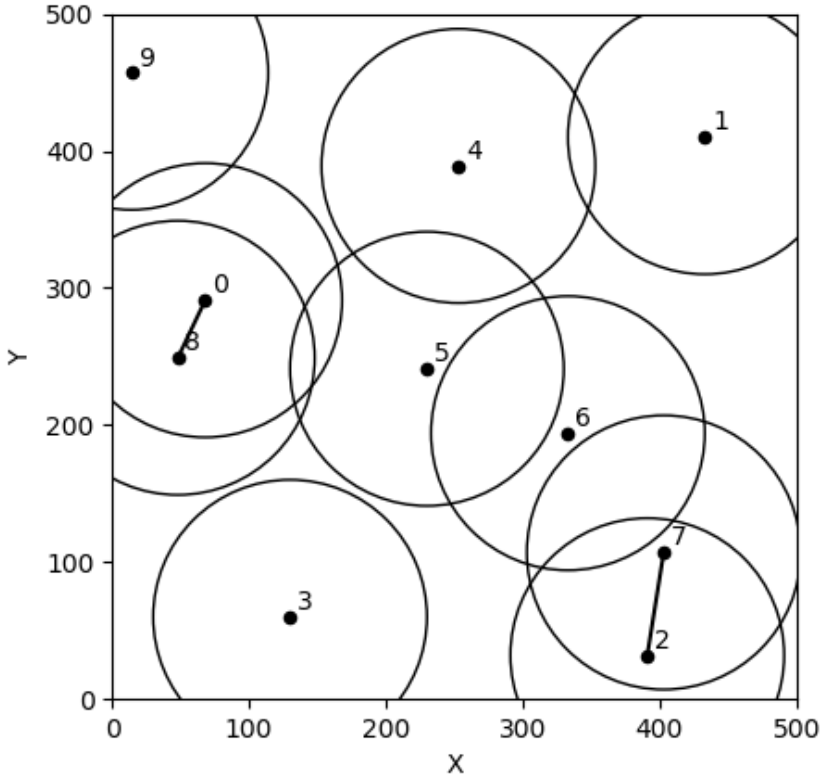
Model Parameter and Threshold Aggregation through Device-to-Device Communication

- WAFL Model Parameter Aggregation

$$W^n \leftarrow W^n + \lambda \frac{\sum_{k \in nbr(n)} (W^k - W^n)}{|nbr(n)| + 1}$$

- Threshold Aggregation

$$\alpha^n \leftarrow \frac{\sum_{k \in nbr(n)} \alpha^k + \alpha^n + \gamma\beta^n}{|nbr(n)| + 1 + \gamma}$$



nbr(n) gives a set of the neighbors of device n.

These algorithms aggregate models and thresholds across distributed devices.

Model Parameter and Threshold Aggregation through Device-to-Device Communication

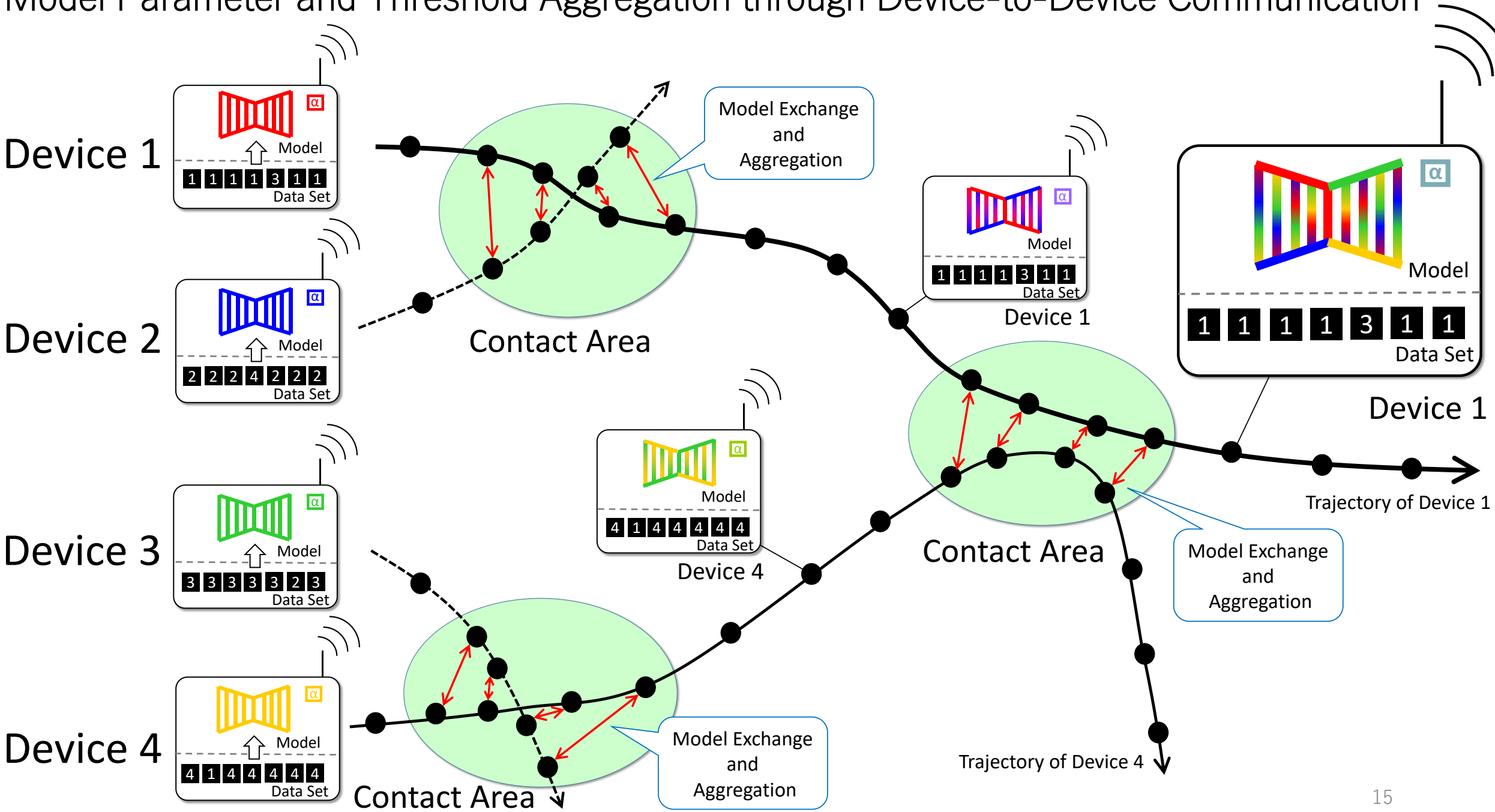
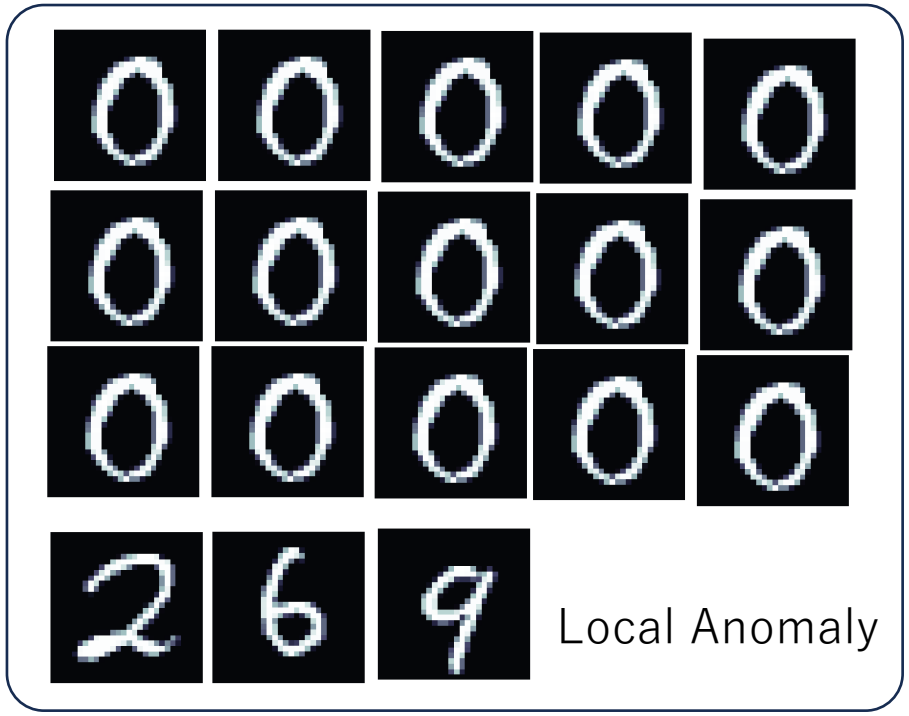


Table of Contents

Data Set Configuration

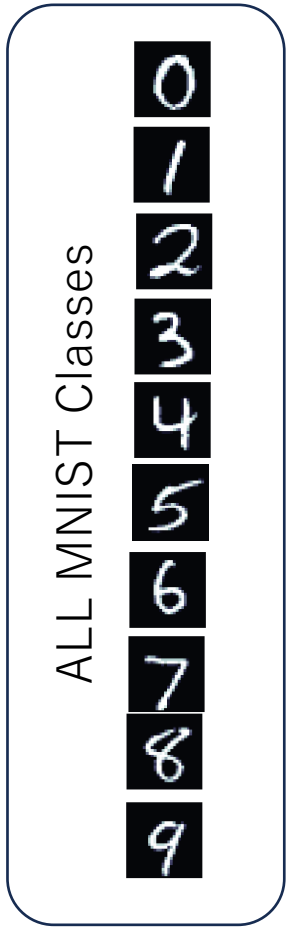
Non-IID Train/Val Dataset @ Device 0



Legitimate Data

Other training dataset for other devices
We considered 10 devices totally.

Test Dataset for all the devices



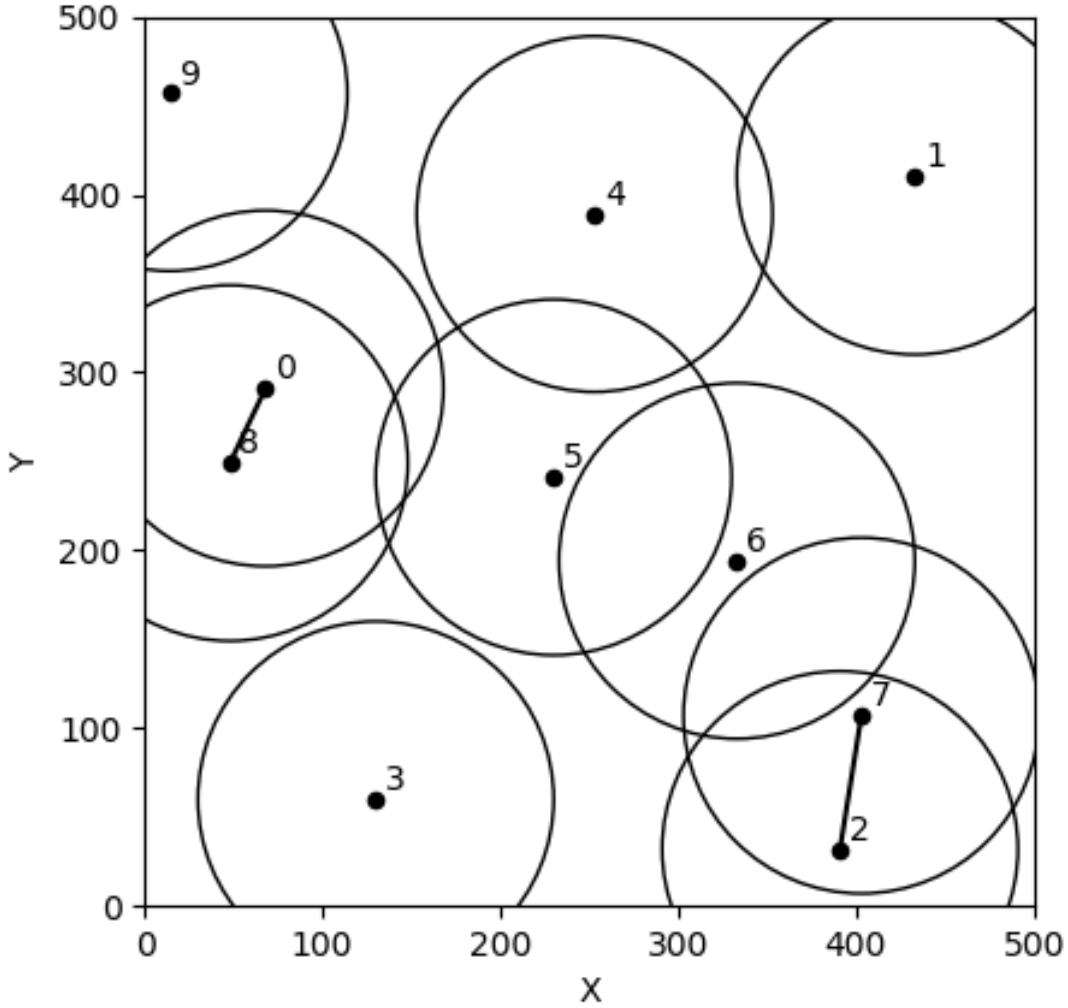
Legitimate **Global Anomaly**

Experimental Setup

- ML model (2D-CNN)

Comp.	Layer (Parameters)
Encoder	Conv2D (i=1, o=8, k=3, p=1) - ReLU - Conv2D (i=8, o=16, k=3, p=1) - BatchNorm2d (16) - ReLU - Conv2D (i=16, o=32, k=3, p=0) - Flatten - ReLU - Linear (i=288, o=128) - ReLU - Linear (i=128, o=64)
Decoder	Linear (i=64, o=128) - ReLU - Linear (i=128, o=288) - ReLU - Unflatten - TransConv2D (i=32, o=16, k=3, p=0, op=0) - BatchNorm2d (16) - ReLU - TransConv2D (i=16, o=8, k=3, p=1, op=1) - BatchNorm2d (8) - ReLU - TransConv2D (i=8, o=1, k=3, p=1, op=1) - Sigmoid

- Mobility Pattern
  - Random Waypoint Mobility (RWP)
- Simulation
  - We carried out the experiment by simulation on a single computer.

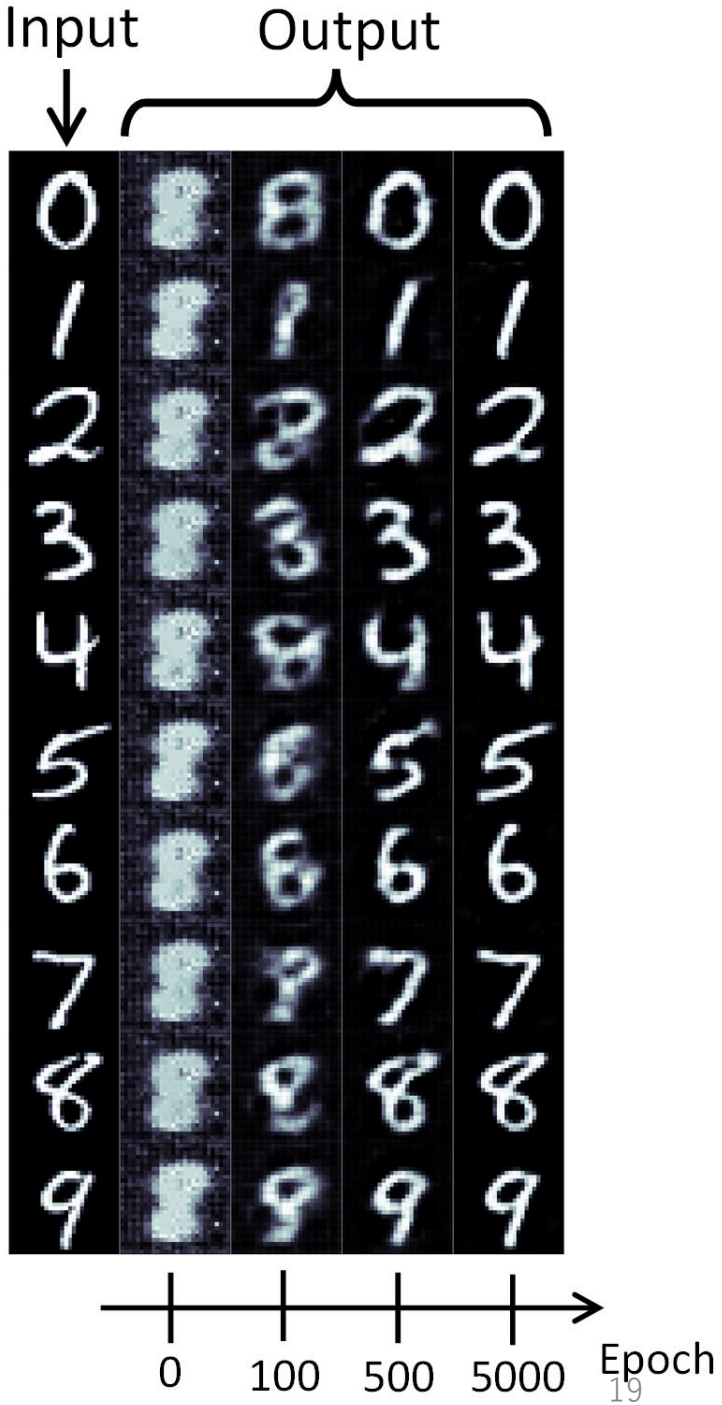


Random Waypoint Mobility

Reconstructions at Device 8 with WAFL In case of Legitimate Inputs

1. WAFL (epoch 0 – after Self-Training)
   the autoencoder gave 8 to any inputs 0 ～ 9.

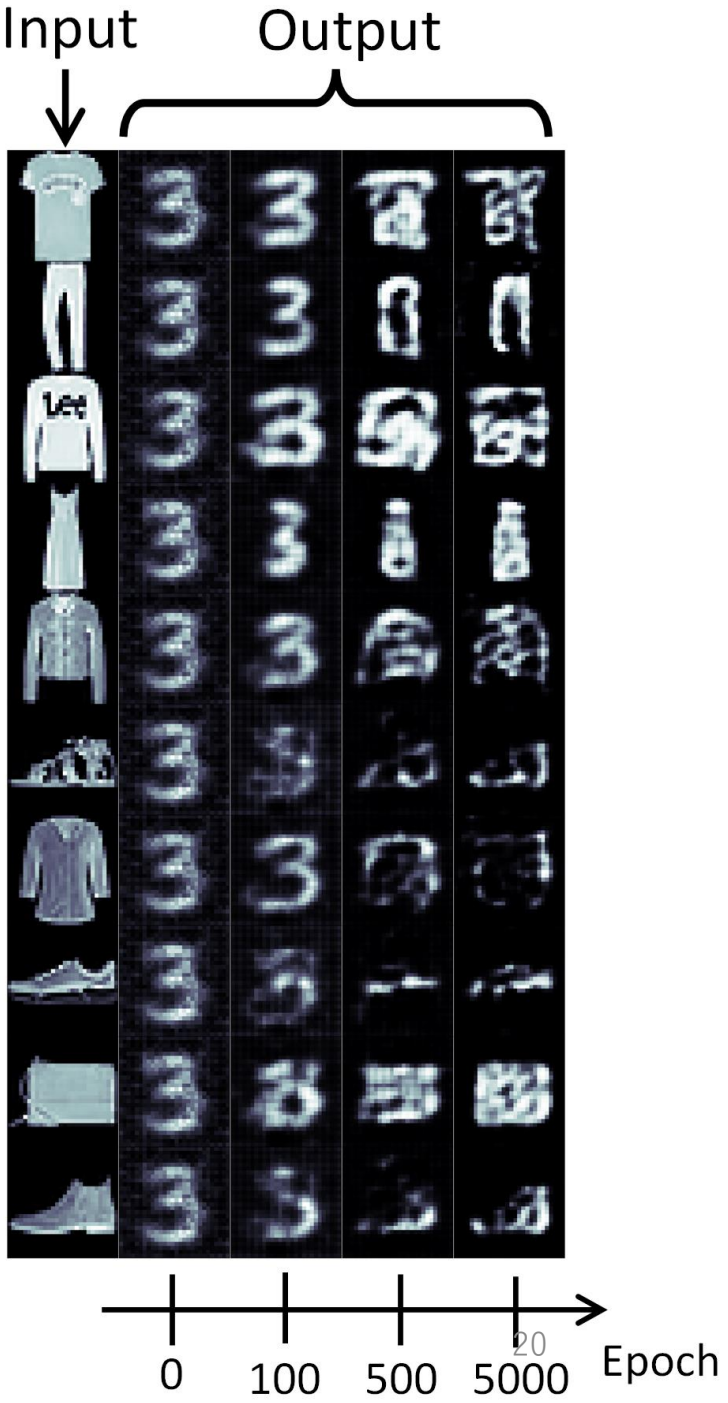
2. As the WAFL training proceeds, WAFL's model aggregation allowed the precise reconstruction of all the legitimate samples (0 ～ 9).



Reconstructions at Device 3 with WAFL In case of Global Anomaly Inputs.

1. WAFL (epoch 0 – after Self-Training)
   the autoencoder gave 3 to any global anomalies.

2. Even though the WAFL training proceeded,
   the autoencoder did not reconstruct the
   global anomaly input (which is succeeded).



Global Anomaly Detection: Performance Overview

with D2D Communication (Our Proposal)

without D2D Communication

Device	WAFL-AE (train without anomaly)					Self-train (without anomaly)				
	TPR				FPR	TPR				FPR
	N	O	F	K		N	O	F	K	
0	100%	16.2%	99.5%	99.5%	0.68%	100%	84.3%	98.1%	99.6%	42.2%
1	100%	5.75%	100%	100%	3.66%	100%	80.7%	98.7%	99.6%	63.5%
2	100%	5.21%	99.5%	99.6%	0.33%	100%	25.3%	97.1%	97.7%	3.06%
3	100%	5.60%	99.6%	99.5%	0.40%	100%	18.3%	97.8%	96.4%	2.50%
4	100%	3.09%	99.8%	99.8%	0.39%	100%	39.6%	99.6%	99.7%	22.1%
5	100%	2.82%	99.7%	99.7%	0.33%	100%	20.9%	99.4%	99.1%	3.54%
6	100%	9.97%	99.8%	99.8%	1.22%	99.8%	52.9%	94.1%	98.8%	24.4%
7	100%	2.43%	99.9%	99.9%	0.48%	100%	41.8%	96.7%	98.4%	16.2%
8	100%	8.97%	99.7%	99.6%	0.38%	99.9%	9.01%	90.2%	94.4%	1.72%
9	100%	3.69%	99.9%	99.9%	0.40%	100%	24.9%	99.3%	98.8%	8.58%
Avg.	100%	6.37%	99.7%	99.7%	0.83%	100%	40.0%	97.1%	98.2%	18.8%

?

Good score

Bad score

Global anomaly can be detected without D2D communication.
(TPR is good)

However, local anomalies was counted as Global anomaly.
(FPR is bad)

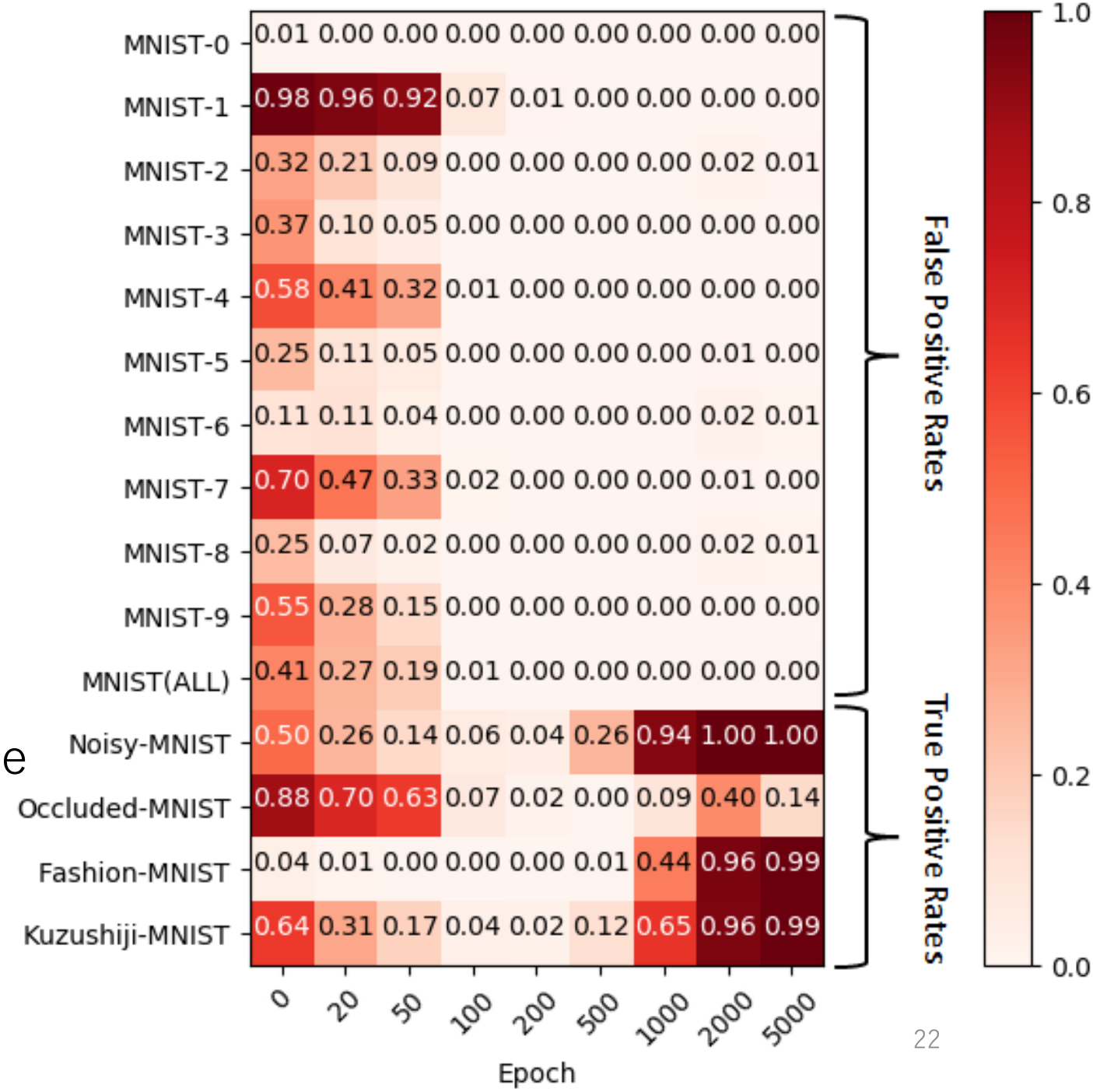
Without D2D, local anomalies was counted as global anomaly.

Anomaly Detection (@Device 0)

1. At Epoch 0 (after Self-Train) many samples except 0 were recognized as anomaly.

2. At Epoch 100, anomaly samples were recognized as legitimate.

3. After Epoch 1000, it could recognize legitimate or anomaly precisely (except Occluded-MNIST).



Latent Space (@Device 4)

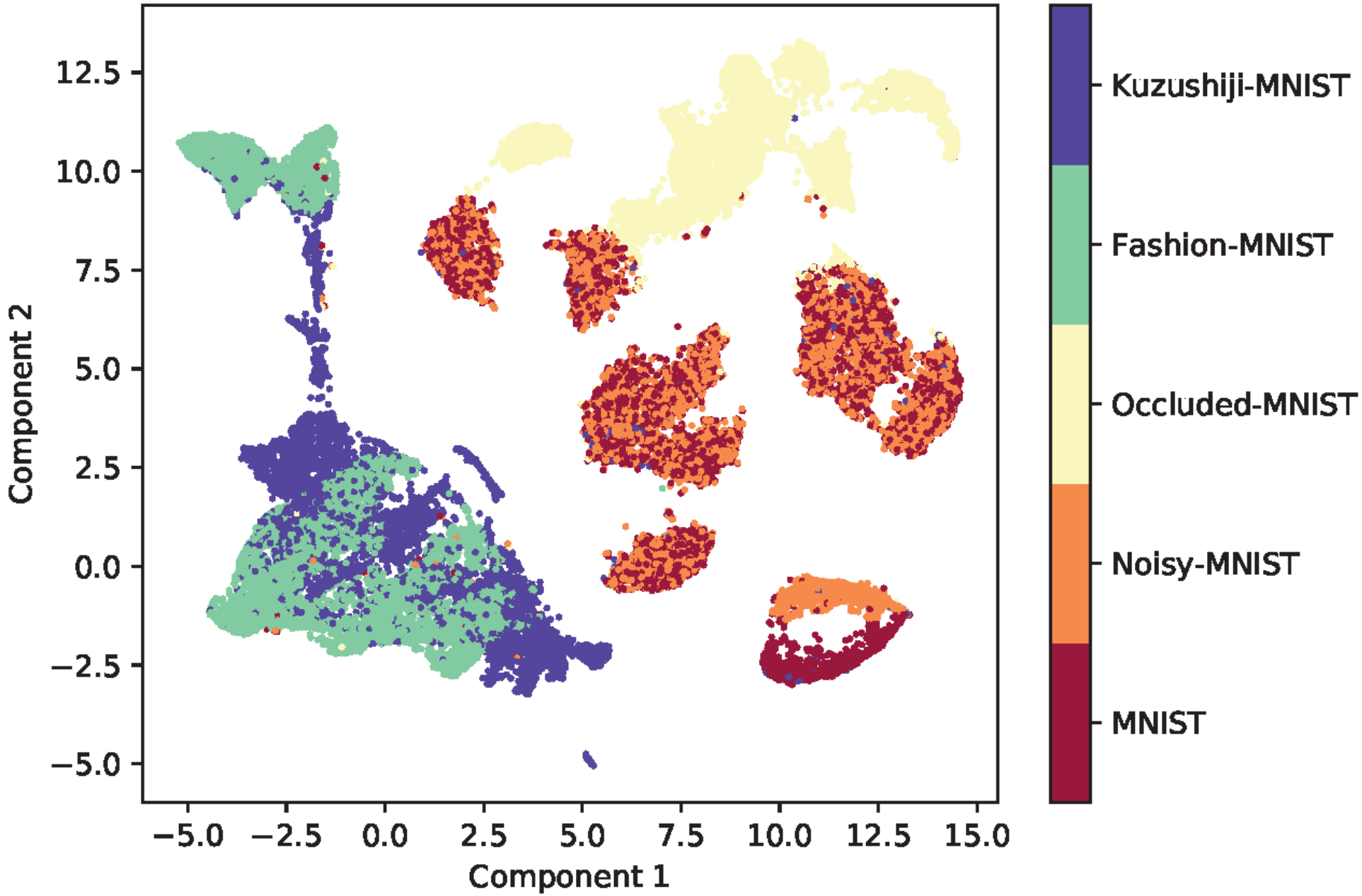


Table of Contents

Conclusion

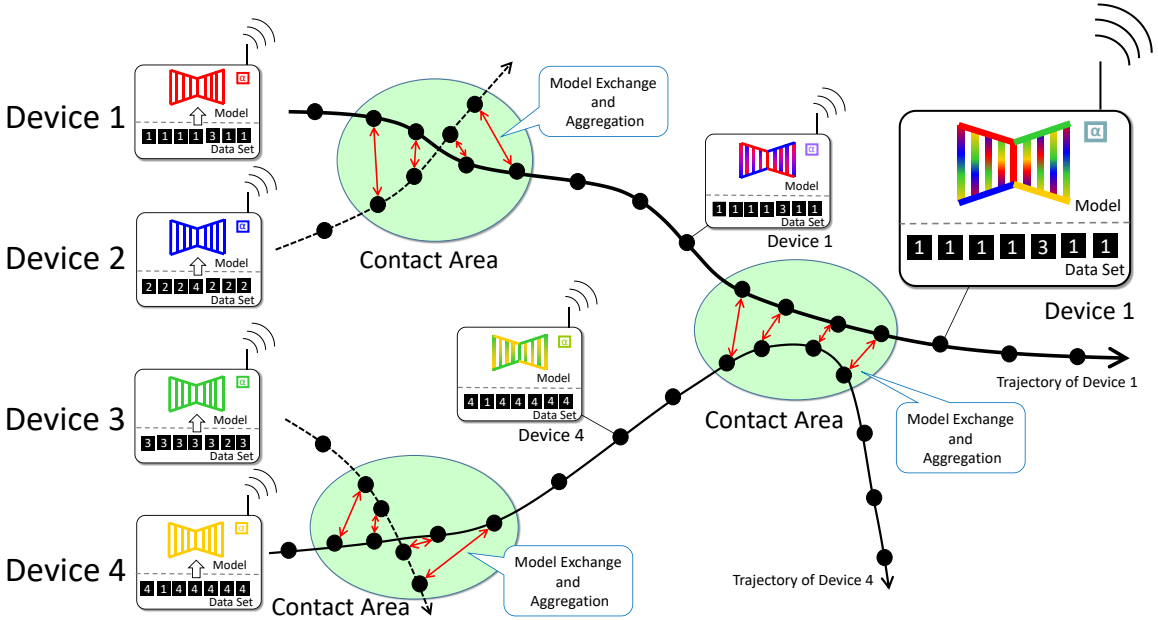
- WAFL-Autoencoder successfully reconstructed legitimate images by aggregating the individually trained autoencoders through device-to-device communication.

- Detection of global anomalies using WAFL-Autoencoder is promising. However, it is still under MNIST-based study.

- More studies will be needed with practical IoT applications.

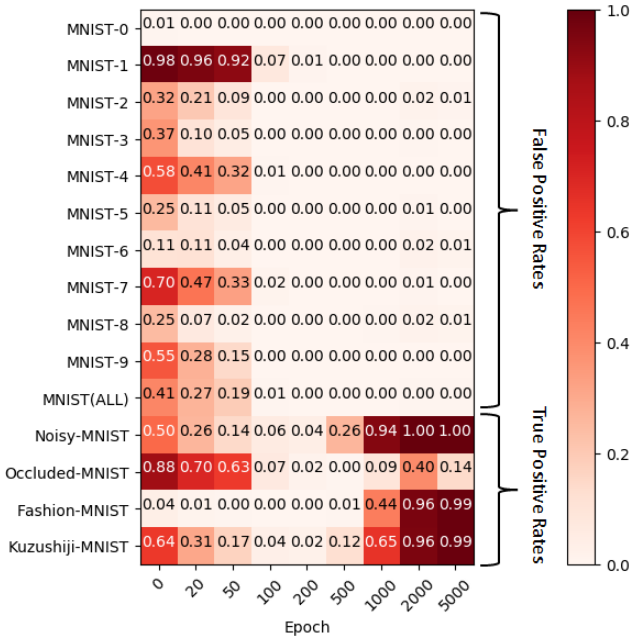
Thank you

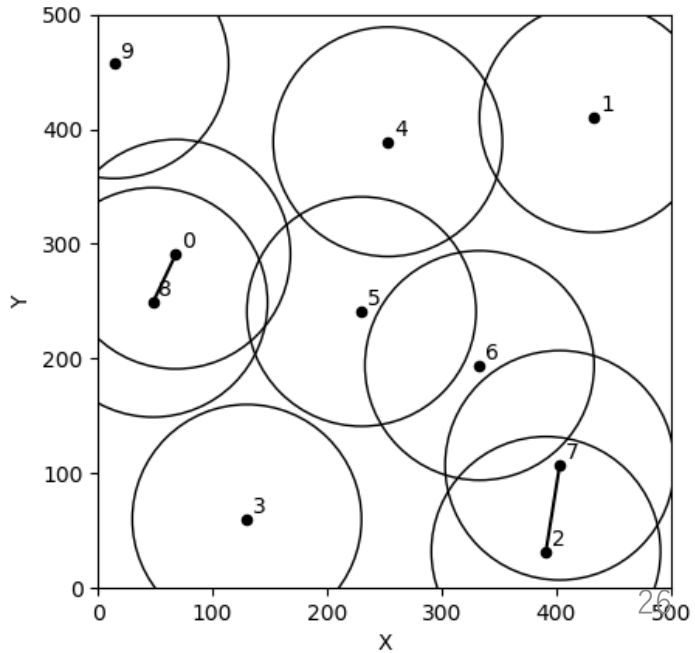

Local Anomaly
(Anomaly for the node, but not for some others)
Device 0
Device 1
Device 2
Device 3
Global Anomaly
(Anomaly for all the nodes)
Our target is “Global Anomaly”


Device 1
Device 2
Device 3
Device 4
Model
Data Set
Model Exchange and Aggregation
Contact Area
Trajectory of Device 1
Trajectory of Device 4
Model Exchange and Aggregation


Input
Output
0
100
500
5000
Epoch


Input
Output
0
100
500
5000
Epoch


MNIST-0
MNIST-1
MNIST-2
MNIST-3
MNIST-4
MNIST-5
MNIST-6
MNIST-7
MNIST-8
MNIST-9
MNIST(ALL)
Noisy-MNIST
Occluded-MNIST
Fashion-MNIST
Kuzushiji-MNIST
False Positive Rates
True Positive Rates
Epoch


X
Y

Appendix

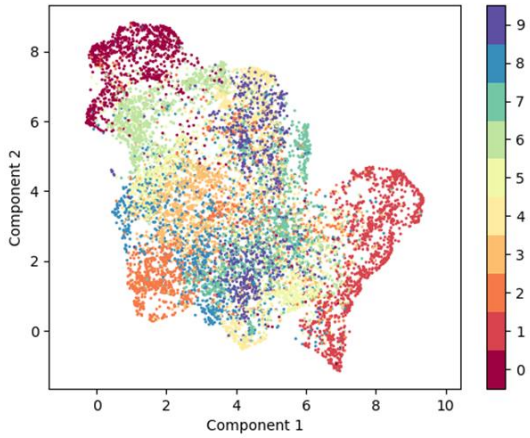
Table 2: True/False positive rates by the developed models and thresholds at epoch 5000 per node. N, O, F, and K indicate Noisy-, Occluded-, Fashion-, and Kuzushiji-MNIST.

Node	WAFL-AE (train without anomaly)					WAFL-AE (train with 1% anomaly)					Self-train (without anomaly)				
	TPR				FPR	TPR				FPR	TPR				FPR
	N	O	F	K		N	O	F	K		N	O	F	K	
0	100%	16.2%	99.5%	99.5%	0.68%	100%	15.9%	92.3%	99.1%	0.54%	100%	84.3%	98.1%	99.6%	42.2%
1	100%	5.75%	100%	100%	3.66%	100%	4.53%	100%	100%	3.27%	100%	80.7%	98.7%	99.6%	63.5%
2	100%	5.21%	99.5%	99.6%	0.33%	100%	5.16%	97.3%	99.7%	0.32%	100%	25.3%	97.1%	97.7%	3.06%
3	100%	5.60%	99.6%	99.5%	0.40%	100%	5.94%	97.9%	99.7%	0.43%	100%	18.3%	97.8%	96.4%	2.50%
4	100%	3.09%	99.8%	99.8%	0.39%	100%	3.00%	99.6%	99.9%	0.37%	100%	39.6%	99.6%	99.7%	22.1%
5	100%	2.82%	99.7%	99.7%	0.33%	100%	2.74%	99.6%	99.9%	0.32%	100%	20.9%	99.4%	99.1%	3.54%
6	100%	9.97%	99.8%	99.8%	1.22%	100%	10.8%	98.7%	99.9%	0.97%	99.8%	52.9%	94.1%	98.8%	24.4%
7	100%	2.43%	99.9%	99.9%	0.48%	100%	2.36%	99.3%	99.9%	0.46%	100%	41.8%	96.7%	98.4%	16.2%
8	100%	8.97%	99.7%	99.6%	0.38%	100%	7.75%	98.0%	99.8%	0.38%	99.9%	9.01%	90.2%	94.4%	1.72%
9	100%	3.69%	99.9%	99.9%	0.40%	100%	3.27%	99.4%	99.9%	0.36%	100%	24.9%	99.3%	98.8%	8.58%
Avg.	100%	6.37%	99.7%	99.7%	0.83%	100%	6.12%	98.2%	99.8%	0.74%	100%	40.0%	97.1%	98.2%	18.8%

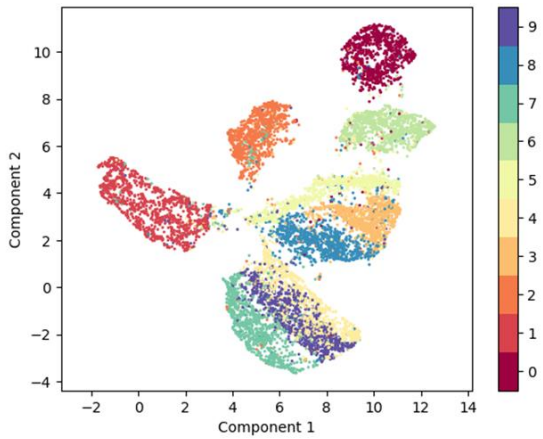
Latent Space (@Device 4)

Classification became clearer

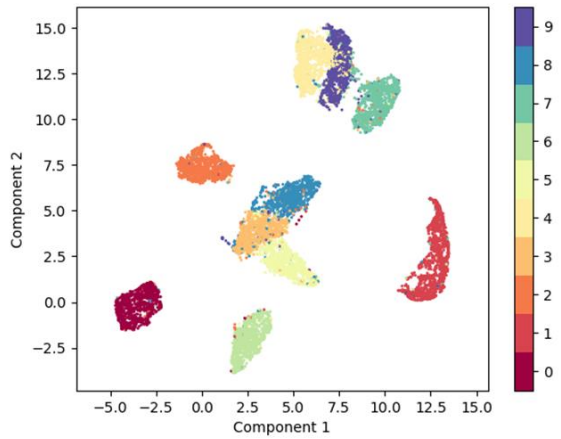
Legitimate Inputs



(a) MNIST, epoch=0

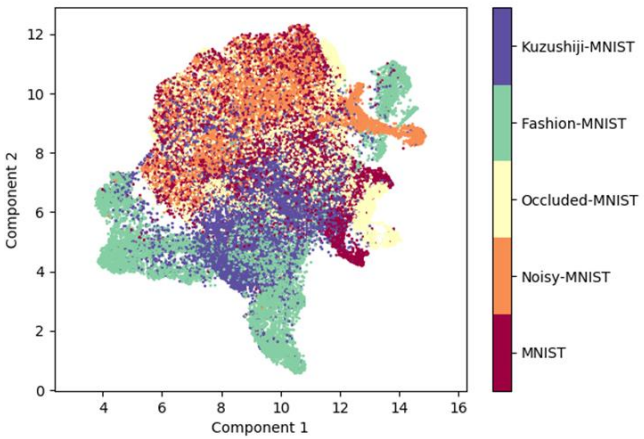


(b) MNIST, epoch=100

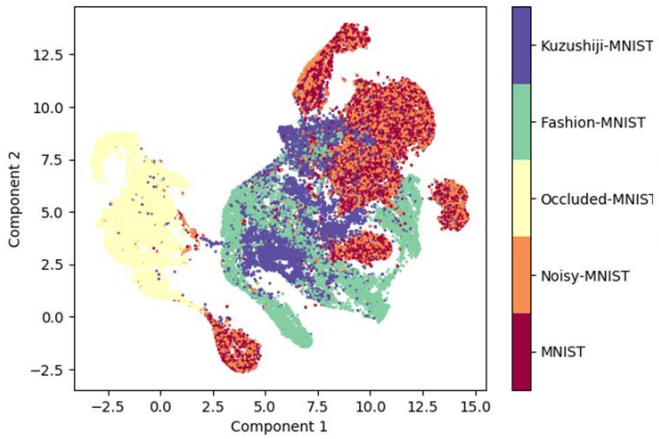


(c) MNIST, epoch=1000

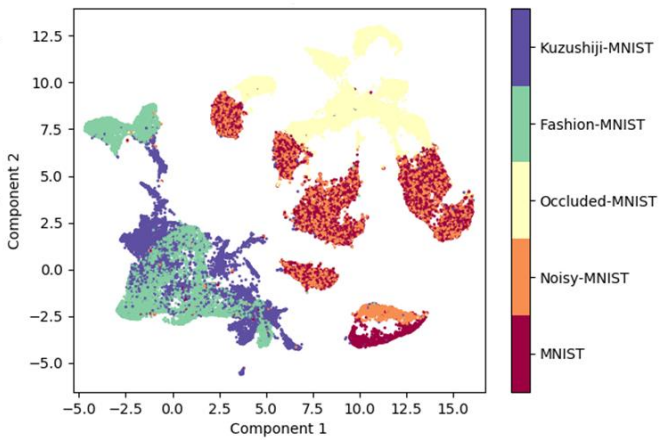
Global Anomaly Inputs



(d) Anomaly, epoch=0



(e) Anomaly, epoch=100



(f) Anomaly, epoch=1000

Anomaly Classes got separated,