Simple Generative Adversarial Network to Generate Three-axis Time-series Data for Vibrotactile Displays

Shotaro Agatsuma (presenter)[1]
Junya Kurogi[2], Satoshi Saga[2] Simona Vasilache[1], Shin Takahashi[1]

[1]University of Tsukuba, [2]Kumamoto University
(Presenter's Email: agatsuma@iplab.cs.tsukuba.ac.jp)

筑波大学
University of Tsukuba

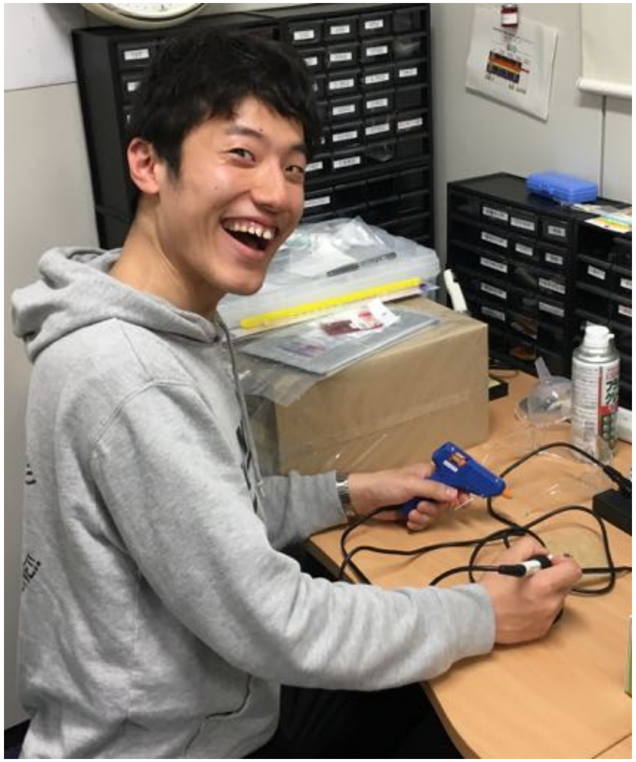
Kumamoto University

IARIA

Self Introduction

- Shotaro Agatsuma
  - University of Tsukuba (Master, 2018~)
    - Intractive Programming Laboratory (https://www.iplab.cs.tsukuba.ac.jp/)
  - Interest : VR (Haptic devices)
    - I have been conducting a collaborative research with Saga Lab in the Kumamoto University
    - Saga Lab

      (http://saga-lab.org/concrete/index.php)

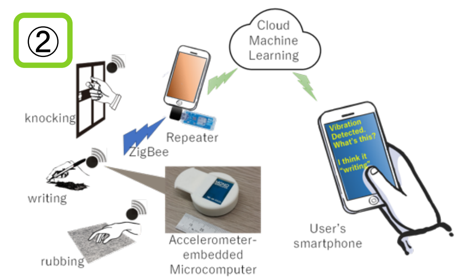


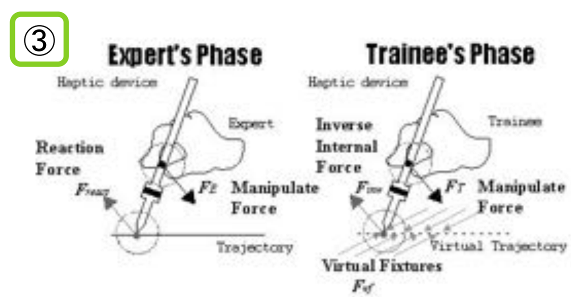
[1] Satoshi Saga and Koichiro Deguchi. Lateral-force-based 2.5-dimensional tactile display for touch screen. In Haptics Symposium (HAPTICS) 2012, pp. 15-22. IEEE, 2012.
[2] Olivier Bau, Ivan Poupyrev, Ali Israr and , Chris Harrison. TeslaTouch: electrovibration for touch surfaces. In Proceedings of the 23nd annual ACM symposium on User interface software and technology, pp. 283-292, ACM, 2010
[3] Heather Culbertson, Joseph M. Romano, Pablo Castillo, Max Mintz, and Katherine J. Kuchenbecker. Refined Methods for Creating Realistic Haptic Virtual Textures from Tool-Mediated Contact Acceleration Data. In Haptics Symposium (HAPTICS) 2012, pp.385-391, IEEE, 2012
[4] Farzan Kalantari, Edward Lank, Yosra Rekik, Laurent Grisoni, and Frédéric Giraud. Determining the Haptic Feedback Position for Optimizing the Targeting Performance on Ultrasonic Tactile Displays. In Haptics Symposium (HAPTICS) 2018, pp.204-209, IEEE, 2018.

Our workgroup

- Saga Lab (http://saga-lab.org/concrete/index.php/en)
  - Main theme of our laboratory is "Expansion of Human Function." We design human-centric interfaces. Research fields are virtual-reality, augmented-reality, human-interface and so on. Currently we work on haptic-based interface technologies.

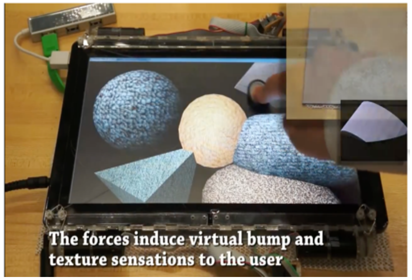
- Researches (right figures : examples)
  ① Tactile Display as a New Media
  ② Tactile Sensor which beyond Human Function
  ③ Haptic Display for Teaching Skills





The methods of presenting tactile sensation

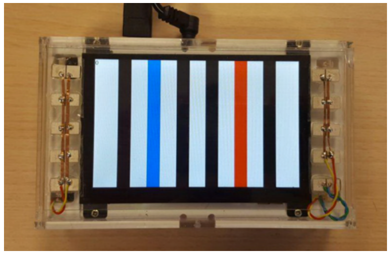
- Tactile displays can present several types of tactile sensation
  - Various kinds of tactile displays have been developed
  - These displays enable users to touch virtual objects



Winding threads [1]

Electrostatic vibration [2]

Voice coil motor [3]



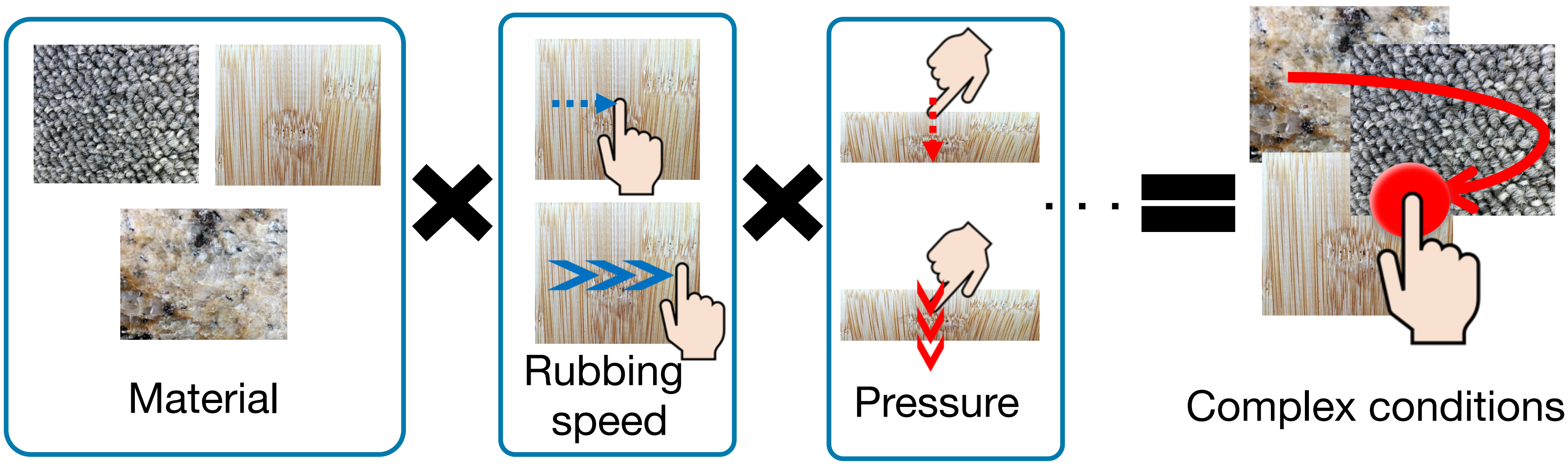
Ultrasonic vibration [4]

- However, it is difficult to reproduce tactile sensations that make users feel like touching real objects
  - Reproducing the physical phenomena of acts of touch is a difficult problem to solve

[1] Satoshi Saga and Koichiro Deguchi. Lateral-force-based 2.5-dimensional tactile display for touch screen. In Haptics Symposium (HAPTICS) 2012, pp. 15-22. IEEE, 2012.
[2] Olivier Bau, Ivan Poupyrev, Ali Israr and , Chris Harrison. TeslaTouch: electrovibration for touch surfaces. In Proceedings of the 23nd annual ACM symposium on User interface software and technology, pp. 283-292, ACM, 2010
[3] Heather Culbertson, Joseph M. Romano, Pablo Castillo, Max Mintz, and Katherine J. Kuchenbecker. Refined Methods for Creating Realistic Haptic Virtual Textures from Tool-Mediated Contact Acceleration Data. In Haptics Symposium (HAPTICS) 2012, pp.385-391, IEEE, 2012
[4] Farzan Kalantari, Edward Lank, Yosra Rekik, Laurent Grisoni, and Frédéric Giraud. Determining the Haptic Feedback Position for Optimizing the Targeting Performance on Ultrasonic Tactile Displays. In Haptics Symposium (HAPTICS) 2018, pp.204-209, IEEE, 2018.

Why is solving this problem is difficult?

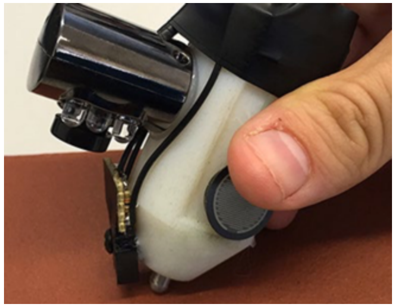
- Acts of touch are affected by numerous conditions of the objects
  - There are a lot of physical conditions of the contactor and contacted objects
    - E.g. material, rubbing speed, pressure, and so on
  - Their combinations become enormous



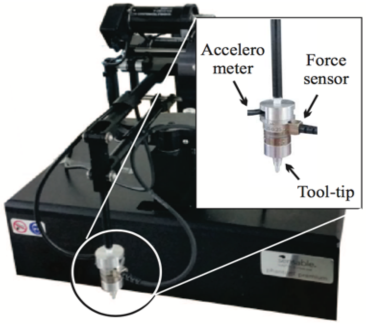
- For presenting high-quality tactile sensations, we should collect and analyze all data under these various combinations of conditions

Research projects about data collection

- Strese et al. [5]
  - They collected six kinds of data from 108 types of textures via a pen-typed sensing device



- Abdulali et al. [6]
  - They collected texture data using the robot arm with accelerometer and pressure sensor

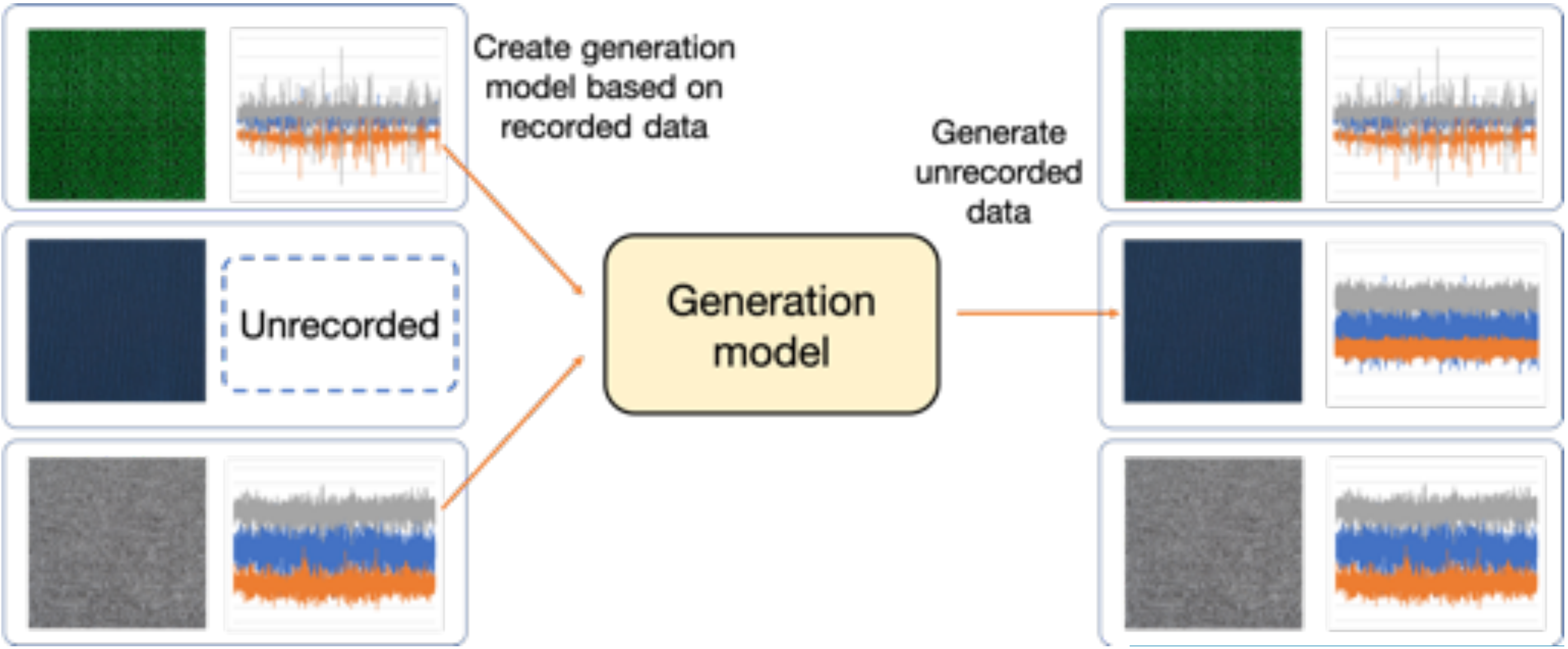


- Problem: Limited conditions
  - Examples：limited textures other than the collected ones, limited rubbing speeds, limited pressures, and so on
  - Collecting all data under numerous combination of conditions is unrealistic

[5]Matti Strese, Yannik Boeck, Eckehard Steinbach “Content-based Surface Material Retrieval”, IEEE World Haptics 2017, pp 352-357. IEEE, 2017
[6]Arsen Abdulali and Seokhee Jeon. Data-driven modeling of anisotropic haptic textures: Data segmentation and interpolation. In International Conference on Human Haptic Sensing and Touch Enabled Computer Applications, pp 228–239. Springer, 2016.

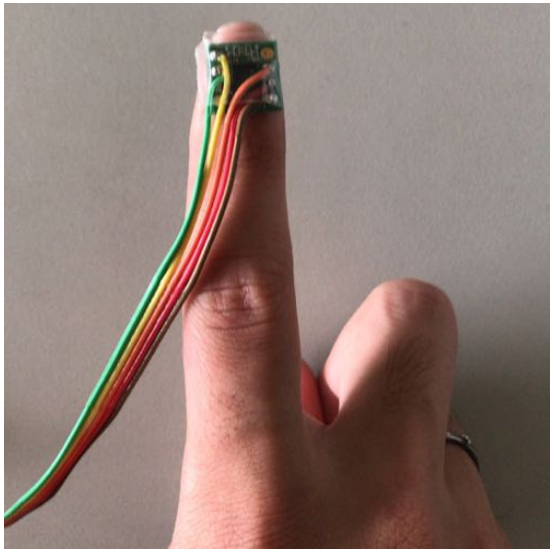
Data generation to solve the problem

- We use machine learning to **generate** alternative data under different conditions from recorded data



Data generation in this study

- **<u>We focused on acceleration data</u>**
  - The acceleration can be used as output signals for some vibrotactile displays
  - The data collection of acceleration is the easy way because there are small and cheap accelerometers



accelerometer

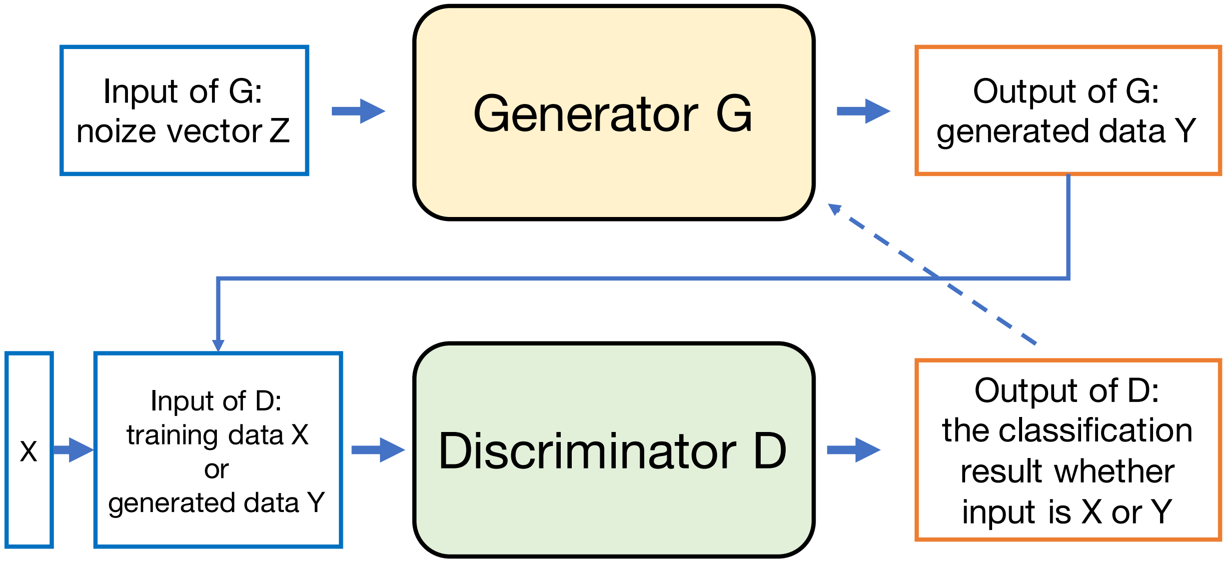
Approach

- We use Generative Adversarial Networks (GANs) [7] for data generation
  - GANs are the methods for generating images
  - GANs have the capability of generating high-quality time-series data such as acceleration
    - With GANs, some researches have succeeded in generating high-quality sounds [8]
  - Depending on the network configuration, GANs can generate unrecorded data [9]

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In Advances in neural information processing systems, pp. 2672–2680. Curran Associates, Inc., 2014.
[8] Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. GANSynth: Adversarial Neural Audio Synthesis. arXiv preprint arXiv:1902.08710, 2019.
9] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks. In Proceedings of the IEEE international conference on computer vision, pp. 2223–2232, 2017.

Framework of GANs

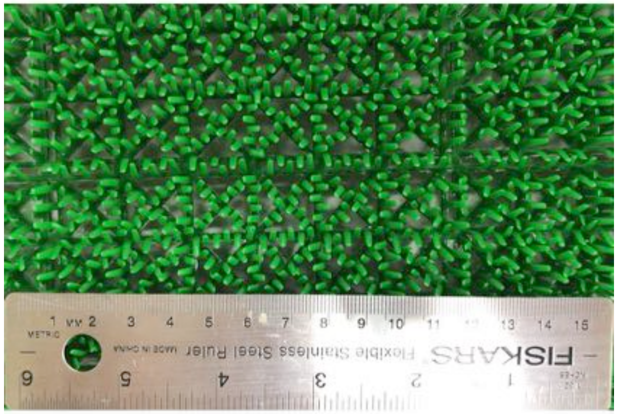
- GANs are composed of two models
  - Generator：generating data
  - Discriminator：classifying the generated data and training data
- The discriminator is trained to classify the generated data and training data accurately
- The generator is trained to generate data that the discriminator cannot classify.
- After the repetitive training of the generator and the discriminator, the generator can generate data that is almost the same as training data



Data generation experiment with GAN

- We held the data generation experiment to research what kind of GAN model is suitable for data generation based on acceleration data
  - C-RNN-GAN [10]
    - GAN + Recurrent Neural Network
  - Deep Convolutional GAN (DCGAN) [11]
    - GAN + Convolutional Neural Network
  - WaveGAN [12]
    - WaveGAN is more specialized in time-series data generation by improving DCGAN

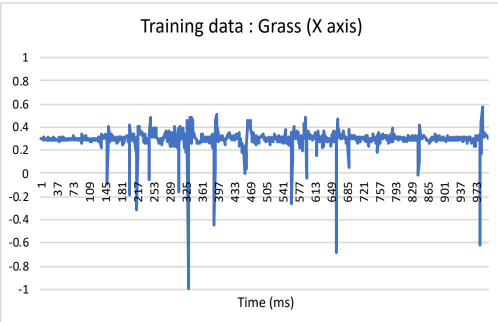
- At first, we generated data based on one kind of data to research whether models can reproduce the data
  - Training data：3-axis acceleration data by rubbing the texture (right figure) with a finger equipped an accelerometer

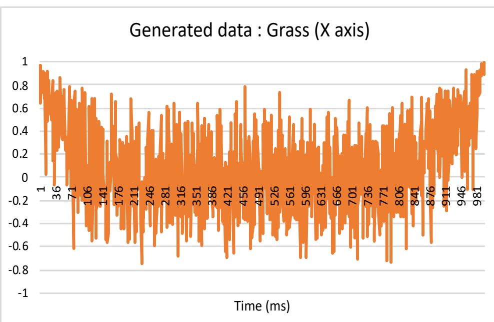


[10] Olof Mogren. C-RNN-GAN: Continuous Recurrent Neural Networks with Adversarial Training. arXiv preprint arXiv:1611.09904, 2016.
[11] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. arXiv preprint arXiv:1511.06434, 2015.
[12] Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial Audio Synthesis. arXiv preprint arXiv:1802.04208, 2018.

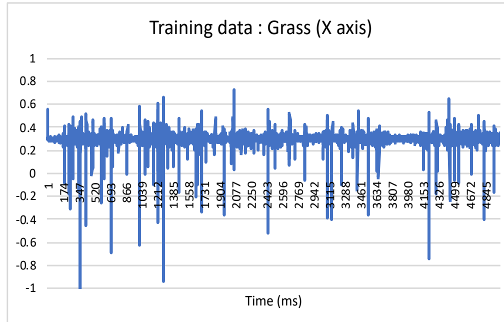
Examples of generated data

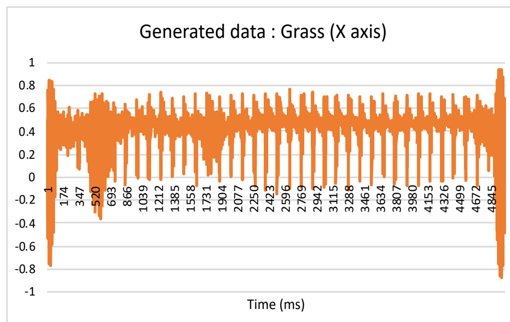
C-RNN-GAN



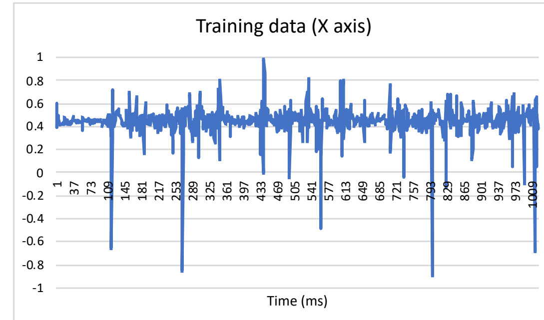


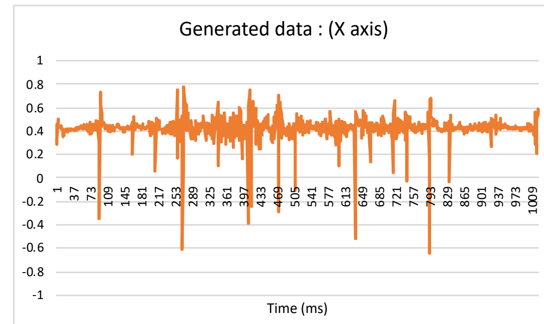
DCGAN





WaveGAN





- We made graphs about training data and generated data
  - Blue : X axis of training data, Orange : X axis of generated data
  - Vertical line : normalized values (-1 to 1) Horizontal lines : times (ms)
- The generated data using WaveGAN is the closest to the training data
  - It seems that WaveGAN is the most suitable to generate the data in these GAN

Data generation experiment with WaveGAN

- WaveGAN can generate many kinds of data or not?
  - Training data : 3-axis acceleration data by rubbing the nine texture with a finger equipped an accelerometer
- Combination with Conditional GAN [13]
  - WaveGAN is unsuitable for generating many types of data. Therefore, we made combined WaveGAN with Conditional GAN
  - Conditional GAN can generate a class-specified data by attaching each class label to the training data when it is trained

[13] Mehdi Mirza and Simon Osindero. Conditional Generative Adversarial Nets. arXiv preprint arXiv:1411.1784, 2014.

Architecture of our GAN

- The architecture is the mostly same as WaveGAN
  - The generator has 14 layers, The discriminator has 17 layers
- Different points
  - We adjusted the input layer for inputting 3-axis acceleration
  - We used Conditional GAN; the input data were combined with the label data
  - We used Leaky ReLU as the activation function in the generator
  - We used the weight initialization method of He et al. [14] to each convolution layer in both models.

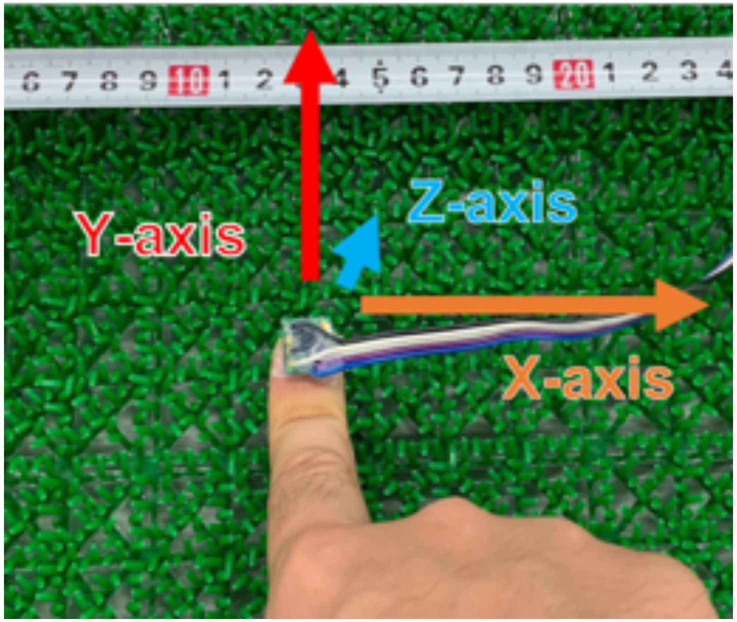
Generator
Input : Uniform(-1,1) + label
Dense
Reshape
LeakyReLU (α=0.2)
Trans Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Trans Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Trans Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Trans Conv2D (Stride= (1, 4))
LeakyReLU (α=0.2)
Trans Conv2D (Stride= (1, 4))
Output : Tanh

Discriminator
Input : Training data or Generated data
Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Phase Shuffle (n = 2)
Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Phase Shuffle (n = 2)
Conv2D (Stride=(1, 4))
Phase Shuffle (n = 2)
LeakyReLU (α=0.2)
Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Phase Shuffle (n = 2)
Conv2D (Stride=(1, 4))
LeakyReLU (α=0.2)
Reshape
Output : Dense

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers: Surpassing Human-level Performance on Imagenet Classification. In Proceedings of the IEEE international conference on computer vision, pp. 1026–1034. IEEE, 2015.

Training data

- We collected 3-axis acceleration data by rubbing the nine textures with a finger equipped an accelerometer
  - The experimenter wore an accelerometer on his index finger
  - He rubbed each texture in one direction for six seconds at a speed of about 5 cm/s under 1 kHz sampling
  - This collection task done 80 times per texture



Other settings of the data generation

■ Hyperparameters

Name	Value
Batch size	64
Phase Shuffle	2
Loss	WGAN-GP
WGAN-GP λ	10
Generator updates per discriminator	2
Optimizer	Adam (α = 1e-4, β_1 = 0.5, β_2 = 0.9)

■ Training period

➢ It took 46 hours for 40 epochs under general windows machine with two GPU (GTX 1080 Ti × 2)

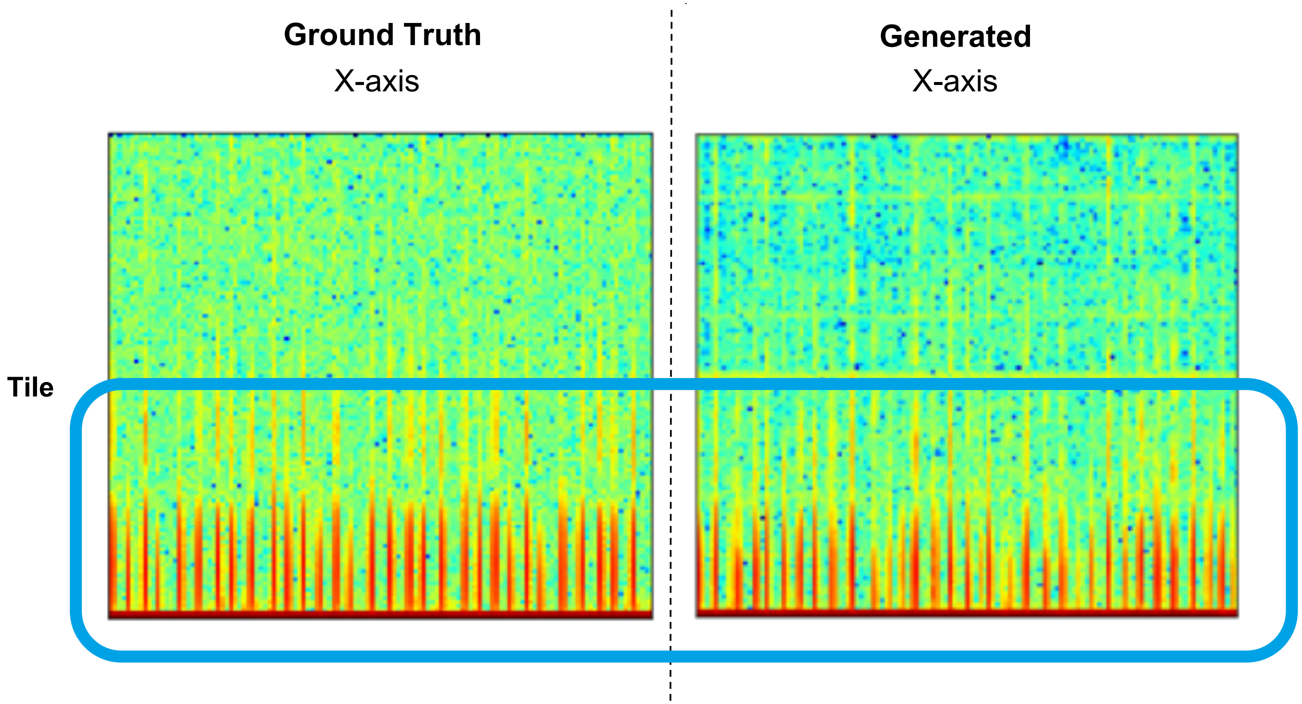
Evaluation methods of generated data

- It is difficult to evaluate whether the generated data reproduce the training data
  - The earlier evaluation methods [15] only can compare several GAN model relatively
- We used two qualitative evaluation methods
  - Comparison of spectrograms between the training data and generated data
    - We validated whether the generated data has characteristics of the training data
  - We presented vibrotactile stimuli to users
    - We validated how realistic the generated vibrotactile stimuli are

[15] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford and Xi Chen, "Improved Techniques for Training GANs," NIPS2016, pp. 2234-2242, 2016

Spectrograms of generated data

- Comparison of spectrograms between the training data and generated data
  - STFT with Hamming window (N = 256), hopsize = 128
  - The generated data appears the same as the training one
- The model can generate data that has characteristics of training data



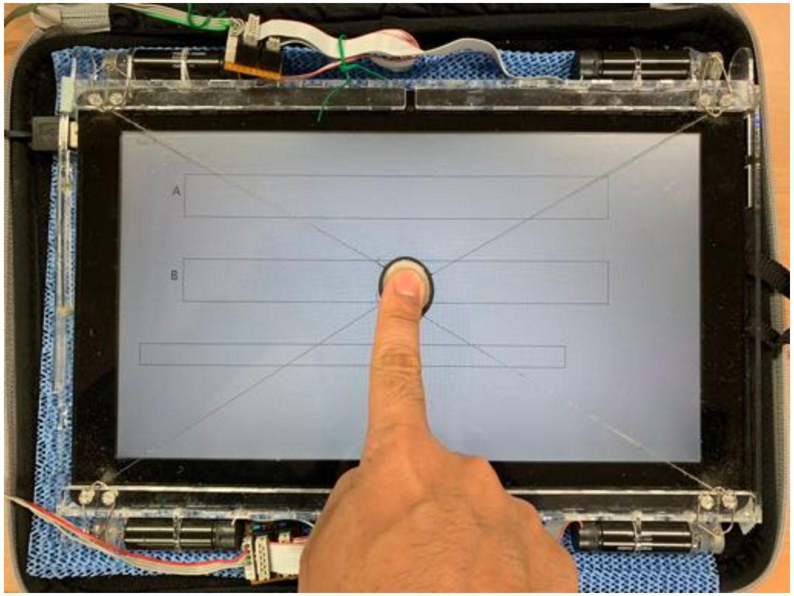
User study

- Evaluate the quality of the generated data
  - We used the collected data for training data
  - We used almost the same task design as Ujitoko's [9]

- Two tasks
  A) Discrimination task
     - Whether the participants could distinguish generated vibrotactile stimuli from training data
  B) Realism evaluation task
     - How realistic the generated vibrotactile stimuli are

Vibrotactile display for the study

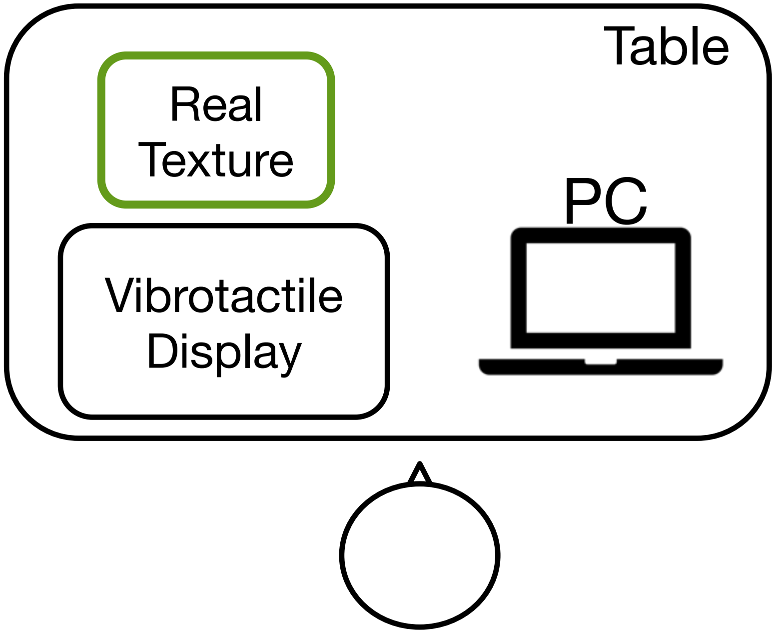
- We used the display proposed by Saga et al. [13]
  - The finger pad is connected to four motors with threads
  - It presents the vibrotactile stimuli by winding the threads
  - It also presents independent vibration of the X and Y-axis



[13] Satoshi Saga and Koichiro Deguchi. Lateral-force-based 2.5 dimensional tactile display for touch screen. In Haptics Symposium 2012, pp. 15–22. IEEE, 2012.

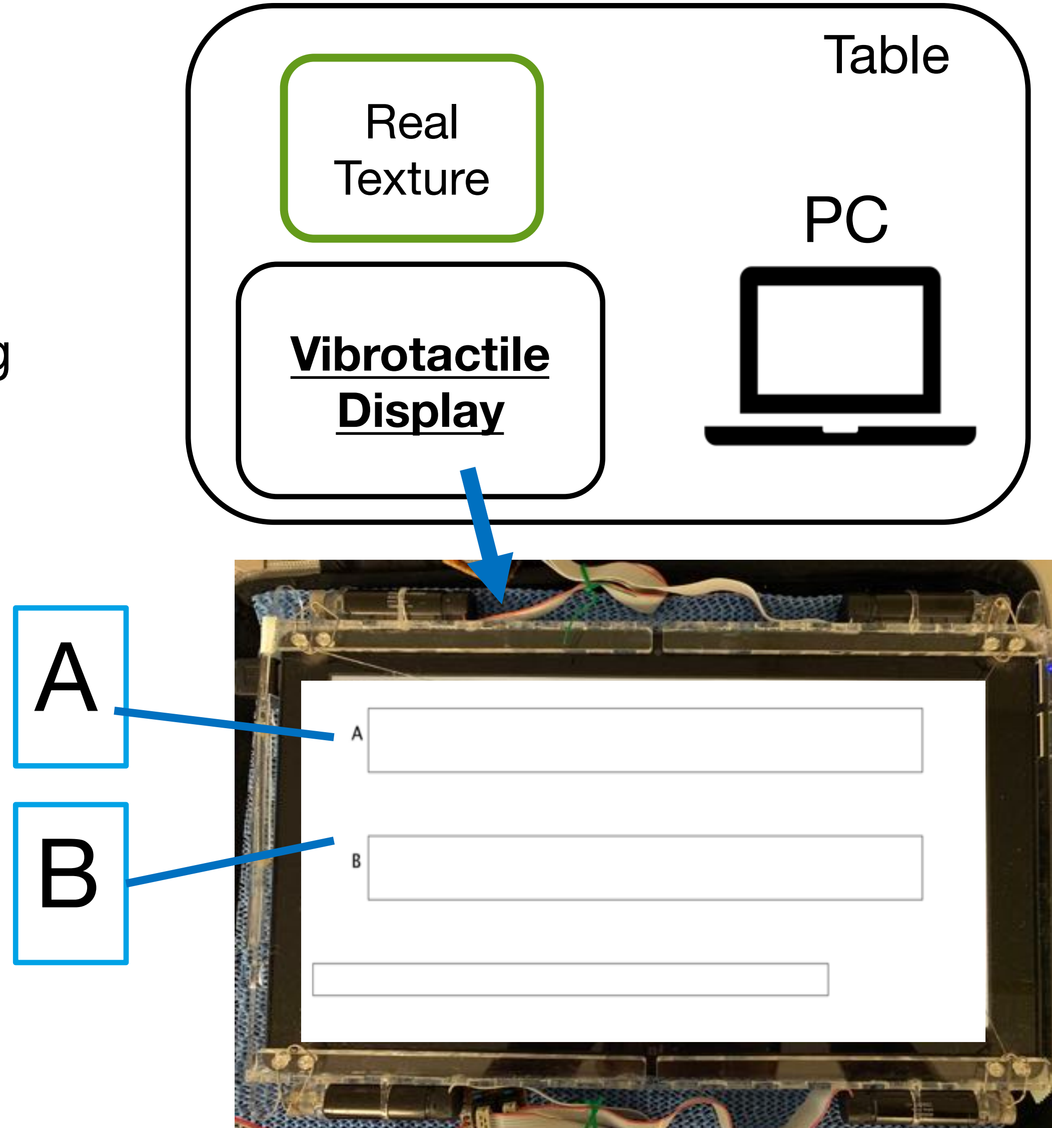
Procedure of the user study

1. Participant had to do the discrimination task
   - Participants distinguished generated data and training data
2. Participant had to do the realism evaluation task
   - Participants were requested to answer the reality of the vibrotactile stimuli from generated data and training data
3. They inputted answers of the tasks into the PC



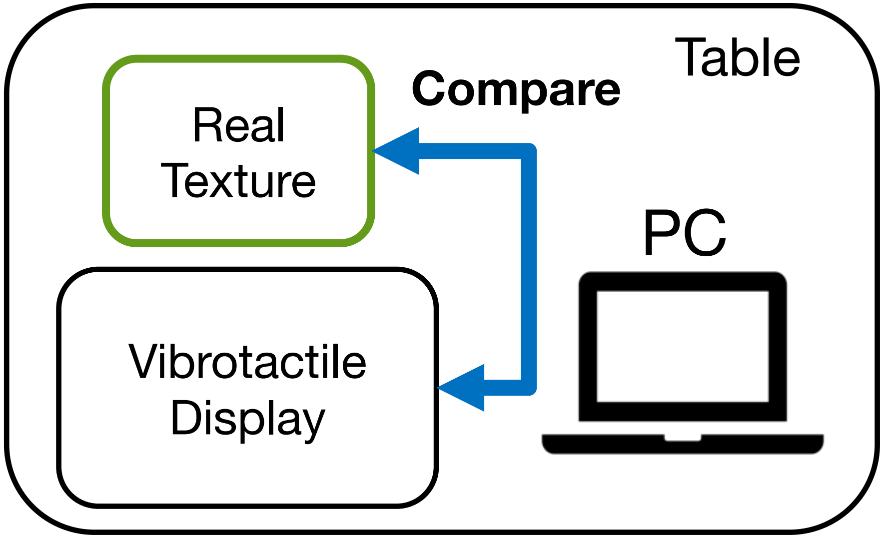
Discrimination task

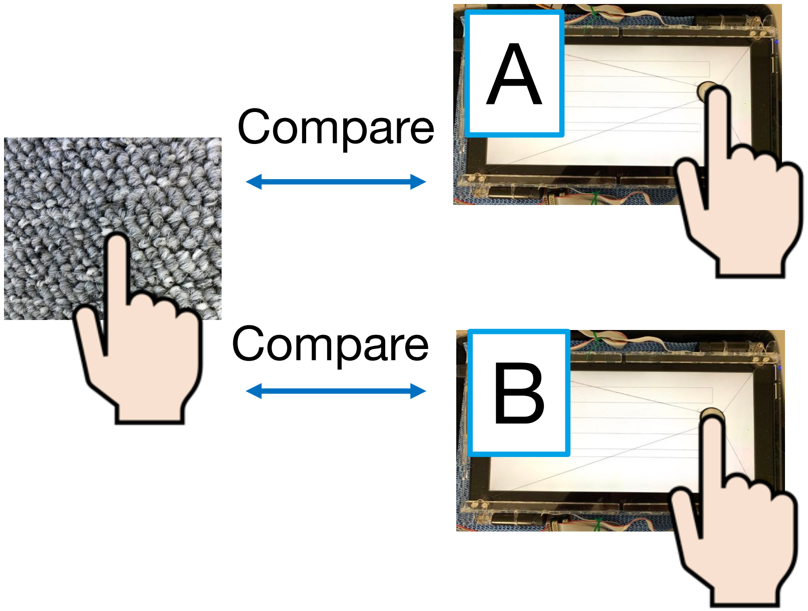
- Participants were requested to rub the predefined paths (A, B)
  - A or B presented a stimulus based on training data, and the other presented a stimulus based on generated data
  - They moved the finger from left to right with 5 cm/s.
- They had to point out which stimulus was generated one



Realism evaluation task

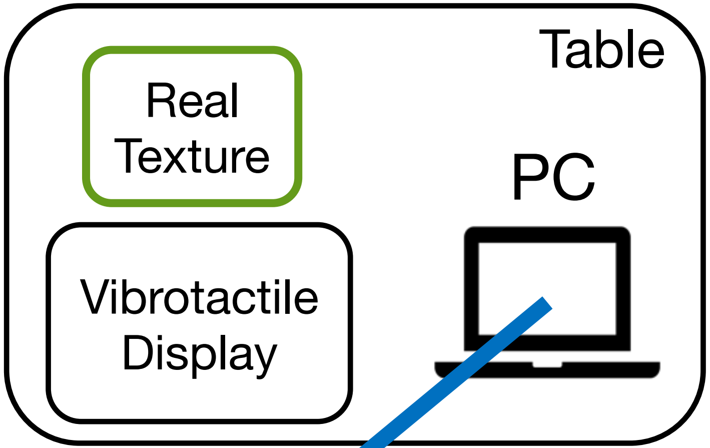
- Participants were requested to compare the real texture with the reproduced one
- Participants were requested to answer the reality of the vibrotactile stimuli from generated data and training data

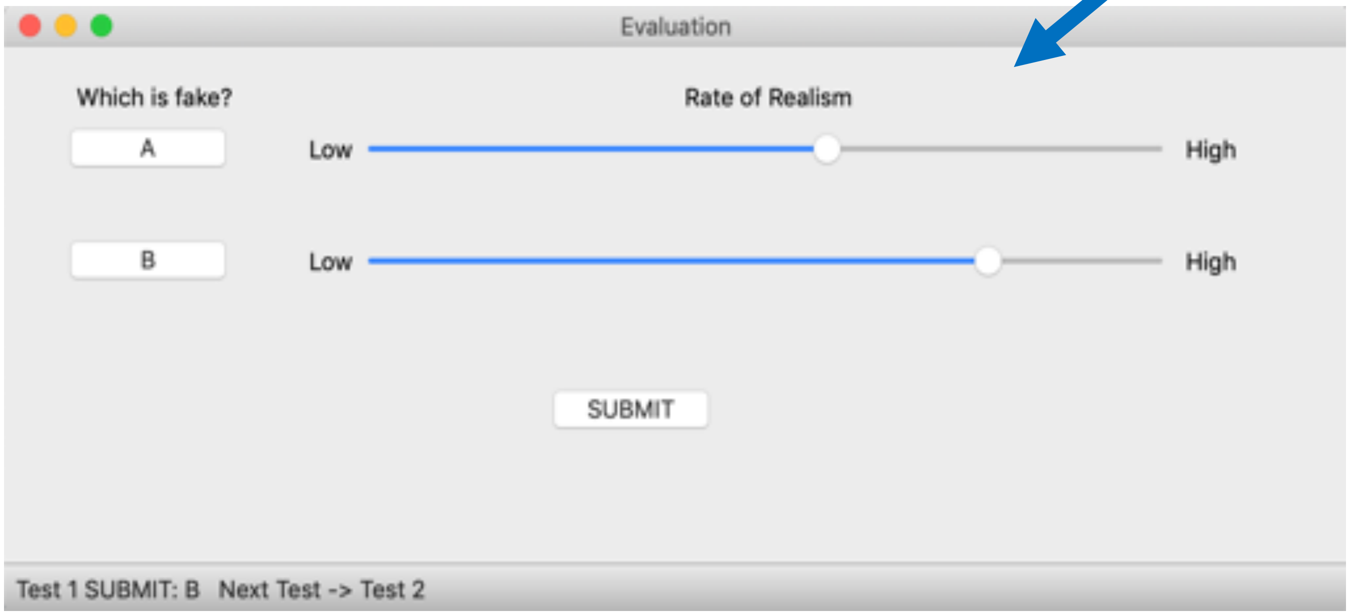




Inputting answers

- Participants inputted the answers into the interface on PC
  - Which stimulus was generated one?
  - The reality of the vibrotactile stimuli
    - The answers are collected by 100 mm of visual analog scale rating

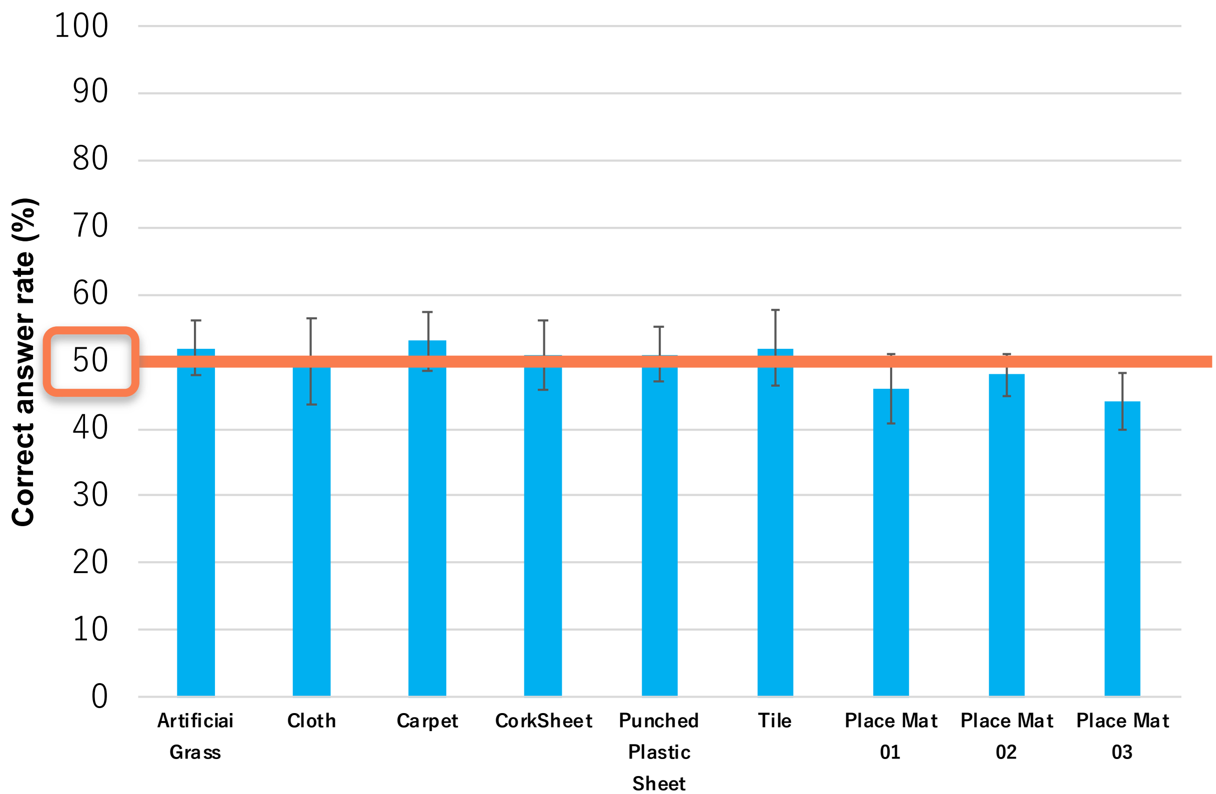




Settings of the tasks

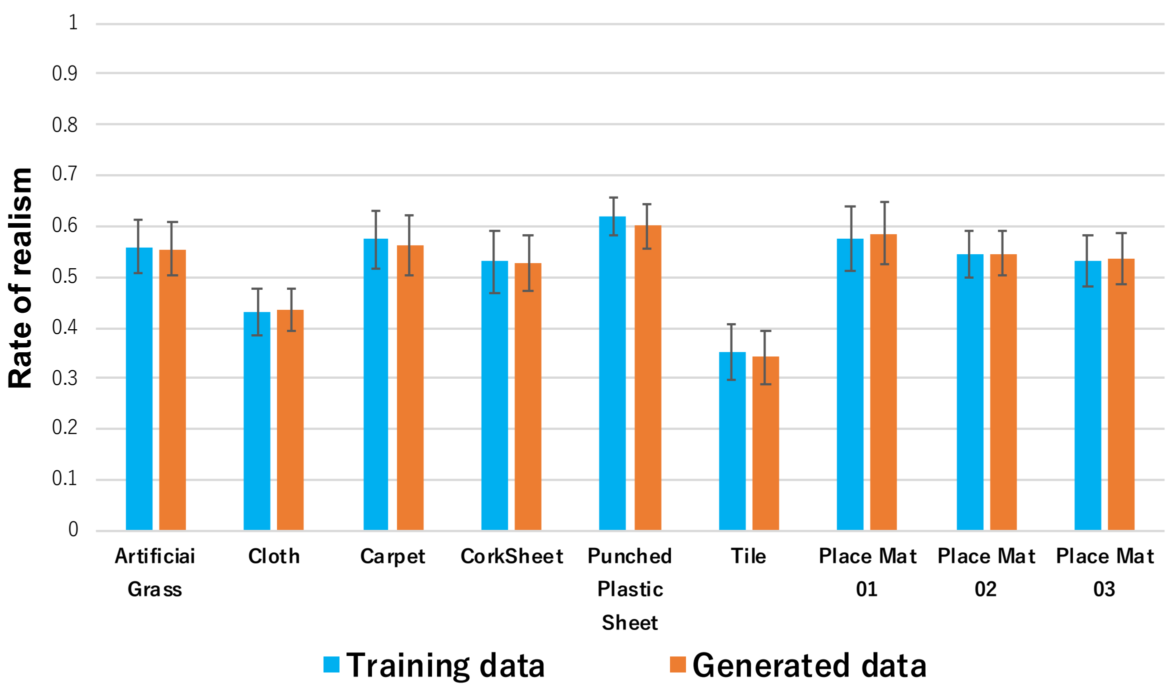
- We held the tasks ten times for each texture
  - There were nine textures
  - We held the studies 90 times for each participant
- Ten participants (eight males and two females 22-24 years old)
- The study took about one hour for each participant

Result of the discrimination task



- The value was almost 50% in all textures
  - **The participants failed to distinguish the training data from the generated data**
  - The model can generate data that is close to acceleration data from training data

Result of the realism evaluation task



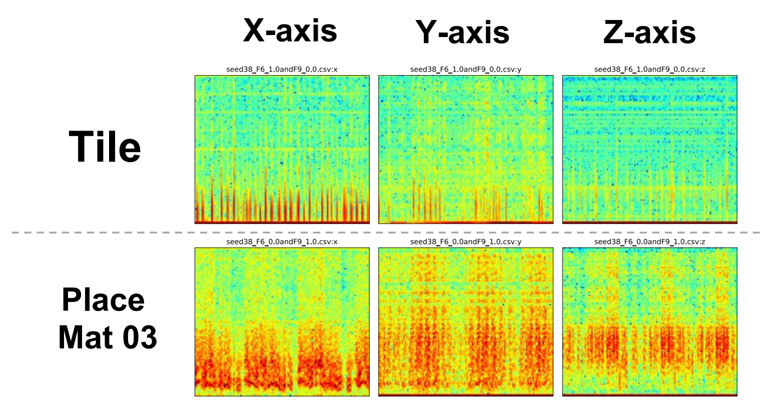
- The value of generated data was almost the same as the training one
  - **We used a Student's paired t-test to each texture condition, and revealed that there were no significant differences**
  - We found that vibrotactile stimuli using the generated data had high reality, like the training data

Unrecorded data generation test

- Result of the user study
  - Our model succeeded in generating the 3-axis time-series data that reproduces recorded data
- Next step: unrecorded data generation test
  - We will evaluate whether the model can generate unrecorded data by controlling the input label
  - For example, we merged two input labels in this test

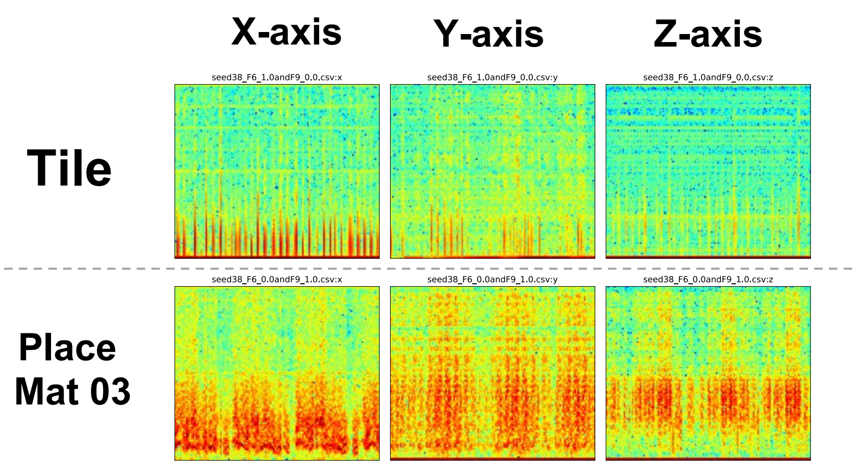
Settings of the test

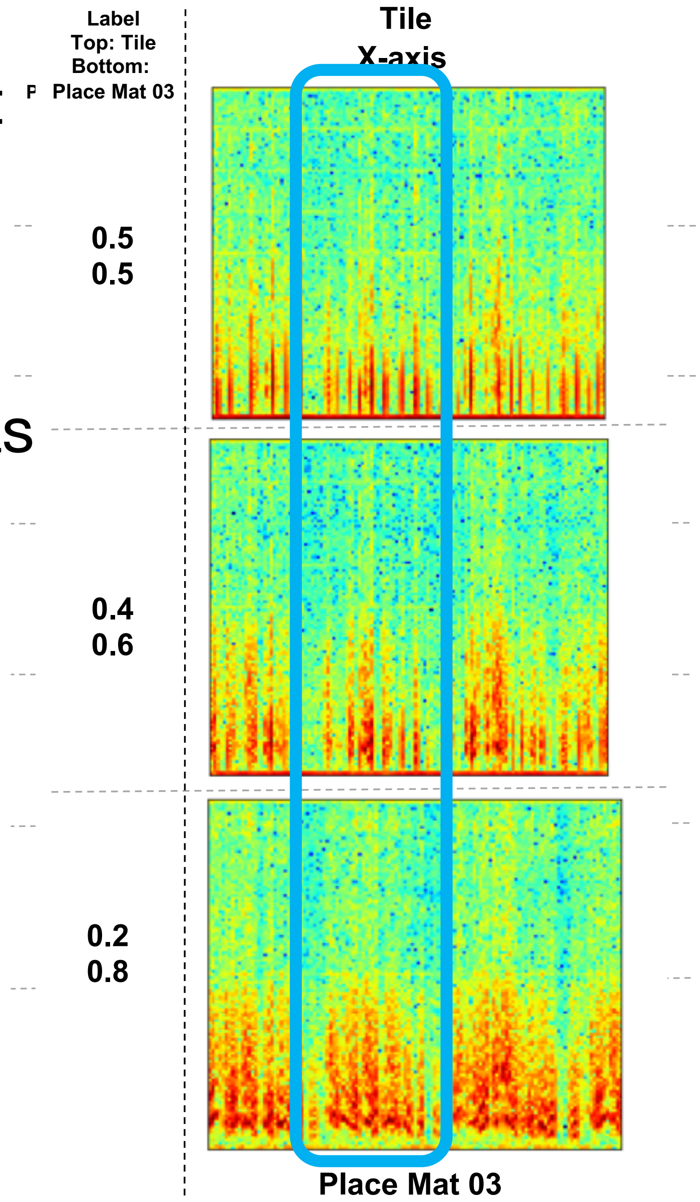
- We used the collected data for training data
- The input label is one-hot vector
  - The vectors have values of either 0 or 1, and their lengths are the same as the number of classes
  - Each class vector has the value of 1 and all others have value of 0
- Before we generated data for “Place Mat 03”, we merged the label for data generation based on “ Tile ” with the “Place Mat 03” input label



Result of the test

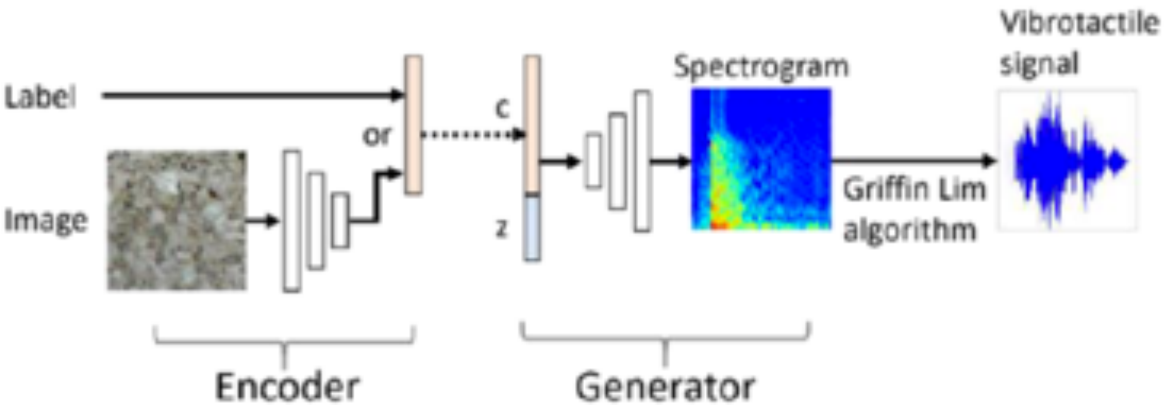
- In the input label, the index of "Place Mat 03" and "Tile" were set between 0.0 to 1.0
- The more the value of the "Place Mat 03" label increased, the more the characteristic of these data was mixed
  - The model is likely to generate unrecorded data if we change the input label





Related work

- Ujitoko et al. [9]
  - They proposed a GANs model that generates time-series data related to texture images (figure)
  - Encoder：encoding texture images into a label vector
  - Generator：generating spectrograms using GANs trained with recoded acceleration and the label
  - The model generates nine types of data for pen-type vibrotactile displays



[9] Yusuke Ujitoko and Yuki Ban. Vibrotactile Signal Generation from Texture Images or Attributes using Generative Adversarial Network. In International Conference on Human Haptic Sensing and Touch Enabled Computer Applications, pp. 25–36. Springer, 2018.

Related work : issues

- **The generated data is used in limited situations**
  - The generated data is 1-axis time-series data for pen-type vibrotactile displays
  - It is difficult to apply the data to the other applications
    - Vibrotactile display that can control multi-axis vibration [1]
    - Analyzing and recognizing the multi-axis vibrotactile signals
  - **Our model can generate 3-axis time-series data**

- **The model has a large number of neural networks**
  - The model requires large computational resources
  - The generator has 27 layers, and the discriminator has 62 layers
  - **Our model is smaller than the model and our model has the same performance as the model**

Conclusion

- We succeeded in generating high-quality time-series data using a simple GAN model
  - The model can generate 3-axis time-series data
  - The training period was about 46 hours
  - From the result of the user study, vibrotactile stimuli of the generated data had high reality, like the stimuli of training data
- The model generates unrecorded data by controlling the input label

Thank you for your kind attention