

Summary of the workshop

Tsuchii Ryotaro

Kuze Laboratory

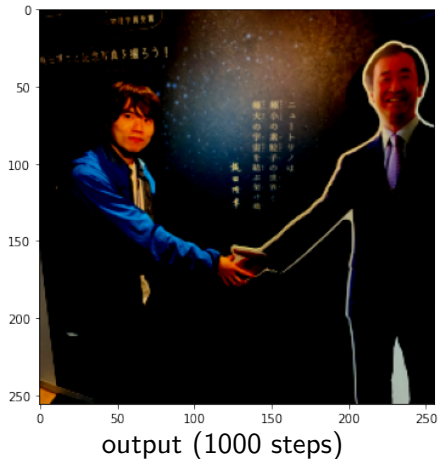
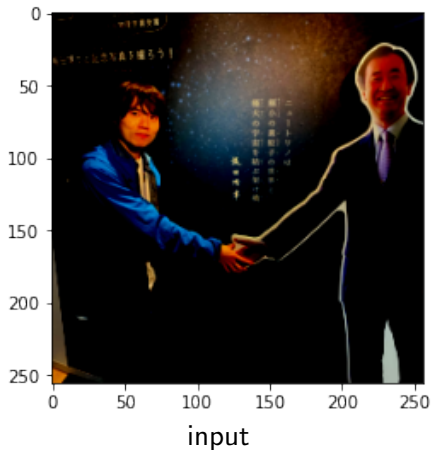
CIDeR-ML Collaboration Meeting
December 14, 2023(JST)

Table of contents

- 1 Abstract
- 2 Picture training
- 3 Training the scattering table
- 4 Summary

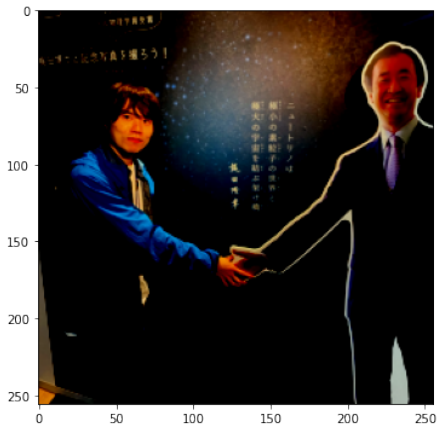
- Goal:
 - to learn how to train SIREN.

Picture training



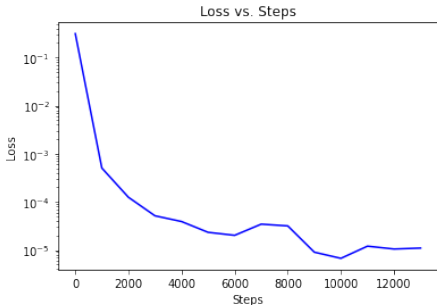
- I trained SIREN with the figure.
 - Input: coords (x, y) (Tensor shape: $[1, 256^2, 2]$).
 - Output: pixel RGBs (r, g, b) (Tensor shape: $[1, 256^2, 3]$).
 - `torch.manual_seed(42)`.

Picture training

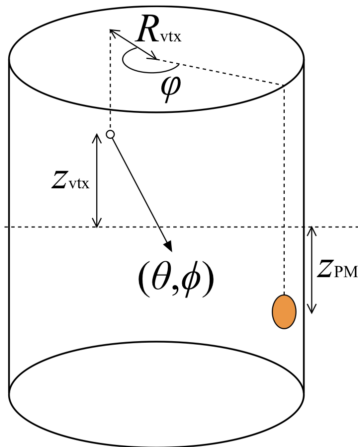


output (1000 steps)

- The loss curve saturated.
- SIREN is trained well.



Training the scattering table (Junjie's notebook)



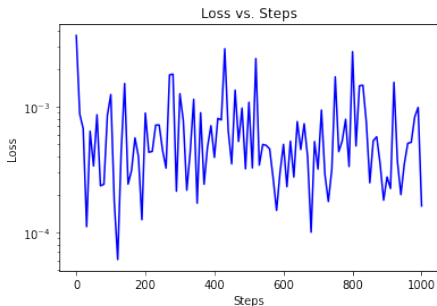
Tobayama-san doctor thesis

- Scattering table (fiTQun):
 - the ratio of the charges from indirect and direct light detected by PMTs.
 - having 6 variables.

$$A(s) = A(x_{PMT}, z_{vtx}, R_{vtx}, \varphi, \theta, \phi)$$

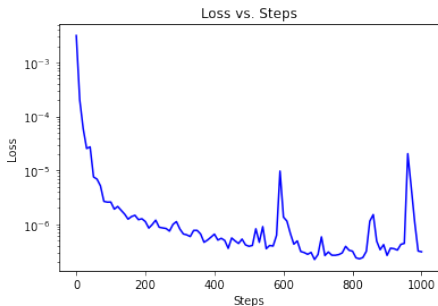
- Assuming rotational symmetry.
- x_{PMT} is corresponded to z_{PMT} .
- Multiply the true by a small factor.
 - it is more convenient to use smaller true during training.
- Normalize the inputs to $[-1, 1]$ by some functions.

Training the scattering table (Junjie's notebook)



loss curve

(ground_truth factor = 10^{-3})



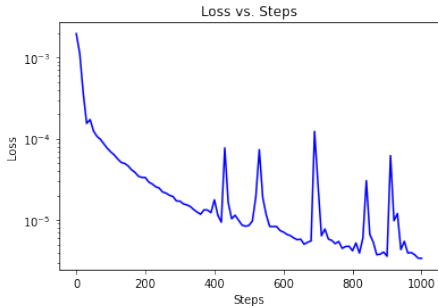
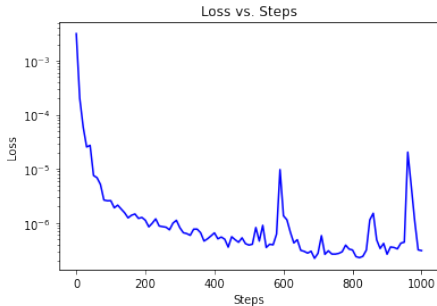
loss curve

(ground_truth factor = 10^{-5})

- I trained SIREN.

- Looks not working well when using factor 10^{-3} .
- `torch.manual_seed(42), random.seed(42)`.
- batch size: 4096.

Training the scattering table (Junjie's notebook)

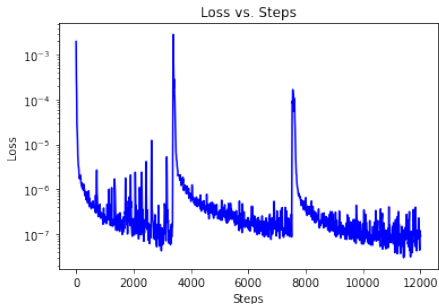


loss curve (normalized sigmoid,
ground_truth = 10^{-5})

loss curve (normalized linearly,
ground_truth = 10^{-5})

- I trained SIREN.
 - I tried to normalize inputs linearly (using $2(v_{\text{input}} - v_{\text{min}})/(v_{\text{max}} - v_{\text{min}}) - 1$).
 - There are some fluctuation.
 - `torch.manual_seed(42)`, `random.seed(42)`.
 - batch size: 4096.

Training the scattering table (Junjie's notebook)

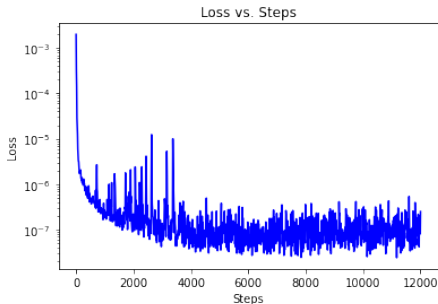


loss curve

(ground_truthfactor = 10^{-5} ,
nomalized by sigmoid,
torch.manual_seed(42),
random.seed(42).)

- Trained for a longer time.
- There are big jumps about once every 4000 steps.

Training the scattering table (Junjie's notebook)

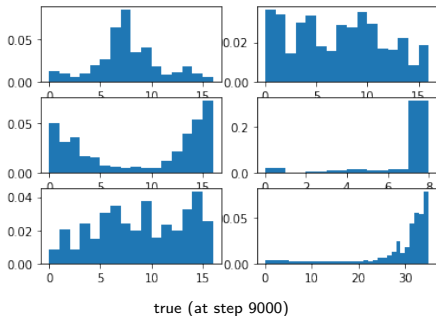


loss curve

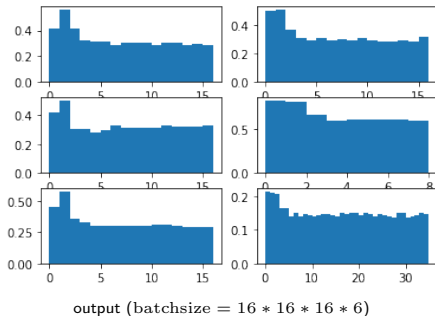
(ground_truthfactor = 10^{-5} ,
nomalized by sigmoid, removed
jump, torch.manual_seed(42),
random.seed(42).)

- I changed the training algorithm to the jumps.
 - ① When detecting jump, go back to the previous saved step.
 - Set the threshold at 100 times the loss of the saved step.
 - Saving SIREN every 50 steps.
 - ② Reduce the learning rate by 1/10.
 - ③ Restart the training.

Training the scattering table (Junjie's notebook)



true (at step 9000)

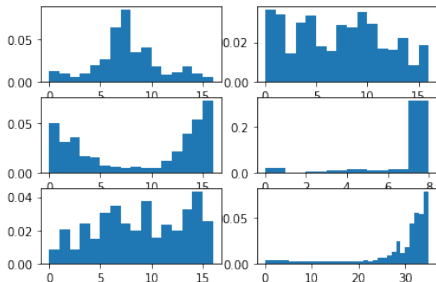


output (batchsize = 16 * 16 * 16 * 6)

- Visualized the tables.

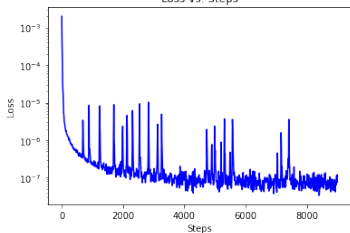
- Create the distribution by projecting the table to each position of same coordinate.
- weighted by each $A(s)$.

Training the scattering table (Junjie's notebook)

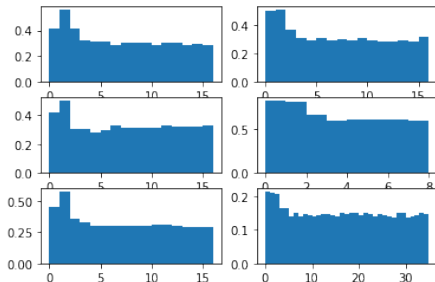


true (at step 9000)

Loss vs. Steps



loss curve (batchsize = 4096 × 6, sigmoid)



output (batchsize = 16 * 16 * 16 * 6)

- Found that the training do not work well.
 - just surating?
 - Something is foundmentally wrong?
 - change some hyperparameters (ω , layer, feature)?

Summary

- Goal: understand how to train SIREN.
- Tried to train SIREN with my picture.
- Tried to train SIREN with the fitQun scattering table.
- I undertood what each parameter means.
- ToDo:
 - Find why the table is not traingd well.
 - create 2D histograms (distribution of true values and SIREN outputs).